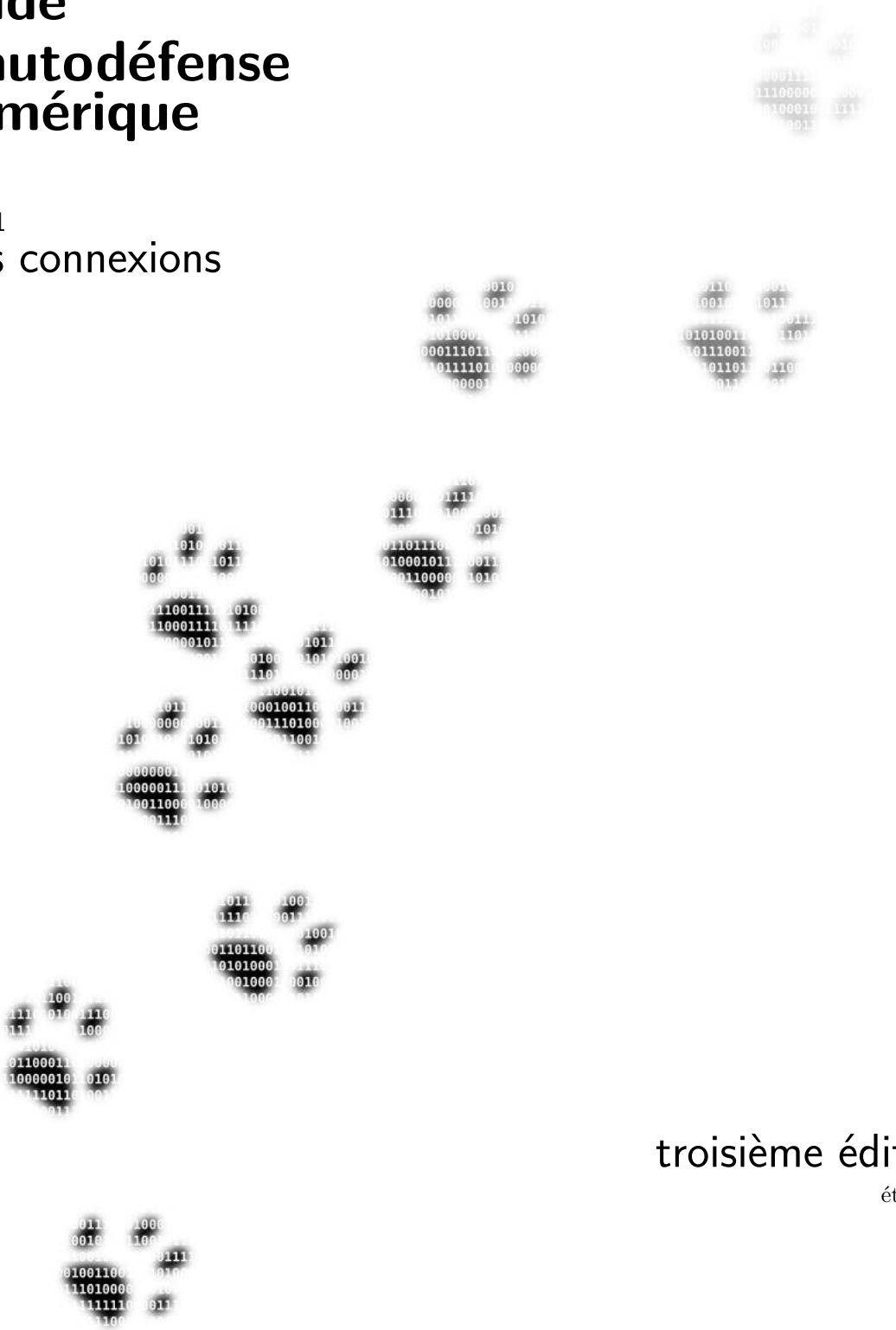
guide d'autodéfense numérique

tome 1

hors connexions



troisième édition

été 2014

ouvrage collectif

Guide d'autodéfense numérique
Tome 1 : Hors connexions
troisième édition

Ouvrage collectif
guide@boum.org

Empreinte OpenPGP :
D487 4FA4 F6B6 88DC 0913
C9FD 326F 9F67 250B 0939

été 2014

Copyleft : cette œuvre est libre, vous pouvez la copier, la diffuser et la modifier selon les termes de la *Licence Art Libre* — http://www.artlibre.org/

Préface

Les revers de la mémoire numérique

De nos jours, les ordinateurs, Internet et le téléphone portable tendent à prendre de plus en plus de place dans nos vies. Le numérique semble souvent très pratique : c'est rapide, on peut parler avec plein de gens très loin, on peut avoir toute son histoire en photos, on peut écrire facilement des textes bien mis en page… mais ça n'a pas que des avantages ; ou en tout cas, ça n'en a pas seulement pour nous, mais aussi pour d'autres personnes qu'on n'a pas forcément envie d'aider.

Il est en effet bien plus facile d'écouter discrètement des conversations par le biais des téléphones portables que dans une rue bruyante, ou de trouver les informations que l'on veut sur un disque dur, plutôt que dans une étagère débordante de papiers.

De plus, énormément de nos informations personnelles finissent par se retrouver publiées quelque part, que ce soit par nous-mêmes ou par d'autres personnes, que ce soit parce qu'on nous y incite — c'est un peu le fond de commerce du *web 2.0*, parce que les technologies laissent des traces, ou simplement parce qu'on ne fait pas attention.

Rien à cacher ?

« *Mais faut pas être parano : je n'ai rien à cacher !* » pourrait-on répondre au constat précédent…

Deux exemples tout bêtes tendent pourtant à montrer le contraire : personne ne souhaite voir ses codes secrets de carte bleue ou de compte *eBay* tomber entre n'importe quelles mains ; et personne non plus n'aimerait voir quelqu'un qui ne lui veut pas du bien débarquer chez lui parce que son adresse a été publiée sur Internet malgré lui…

Mais au-delà de ces bêtes questions de défense de la propriété privée, la confidentialité des données devrait être *en soi* un enjeu.

Tout d'abord, parce que ce n'est pas nous qui jugeons de ce qu'il est autorisé ou non de faire avec un ordinateur. Des personnes arrêtées pour des activités numériques qui ne plaisaient pas à leur gouvernement croupissent en prison dans tous les pays du monde — pas seulement en Chine ou en Iran.

De plus, ce qui est autorisé aujourd'hui, comment savoir ce qu'il en sera demain ? Les gouvernements changent, les lois et les situations aussi. Si on n'a pas à cacher aujourd'hui, par exemple, la fréquentation régulière d'un site web militant, comment savoir ce qu'il en sera si celui-ci se trouve lié à un processus de répression ? Les traces *auront été laissées* sur l'ordinateur… et pourraient être employées comme élément à charge.

Enfin et surtout, à l'époque des sociétés de contrôles de plus en plus paranoïaques, de plus en plus résolues à traquer la subversion et à voir derrière chaque citoyen un

terroriste en puissance qu'il faut surveiller en conséquence, se cacher devient en soi un enjeu *politique*, ne serait-ce que pour mettre des bâtons dans les roues de ceux qui nous voudraient transparents et repérables en permanence.

Quoi qu'il en soit, beaucoup de gens, que ce soient les gouvernants, les employeurs, les publicitaires ou les flics [1], ont un intérêt à obtenir l'accès à nos données, surtout au vu de la place qu'a pris l'information dans l'économie et la politique mondiales.

Tout ça peut amener à se dire que nous n'avons pas envie d'être contrôlables par quelque « Big Brother » que ce soit. Qu'il existe déjà ou que l'on anticipe son émergence, le mieux est sans doute de faire en sorte qu'il ne puisse pas utiliser, contre nous, tous ces merveilleux outils que nous offrent — ou que lui offrent — les technologies modernes.

Aussi, *ayons tous quelque chose à cacher, ne serait-ce que pour brouiller les pistes !*

Comprendre pour pouvoir choisir

Ce guide se veut une tentative de décrire dans des termes compréhensibles l'intimité (ou plutôt son absence) dans le monde numérique ; une mise au point sur certaines idées reçues, afin de mieux comprendre à quoi on s'expose dans tel ou tel usage de tel ou tel outil. Afin, aussi, de pouvoir faire le tri parmi les « solutions », toutes plus ou moins dangereuses si l'on ne se rend pas compte de ce contre quoi elles ne protègent pas.

À la lecture de ces quelques pages, on pourra avoir le sentiment que rien n'est vraiment sûr avec un ordinateur ; et bien, c'est vrai. Et c'est faux. Il y a des outils et des usages appropriés. Et souvent la question n'est finalement pas tant « doit-on utiliser ou pas ces technologies ? », mais plutôt « quand et comment les utiliser (ou pas) ? »

Prendre le temps de comprendre

Des logiciels simples d'utilisation meurent d'envie de se substituer à nos cerveaux… s'ils nous permettent un usage facile de l'informatique, ils nous enlèvent aussi prise sur les bouts de vie qu'on leur confie.

Avec l'accélération des ordinateurs, de nos connexions à Internet, est arrivé le règne de l'instantanéité. Grâce au téléphone portable et au Wi-Fi, faire le geste de décrocher un téléphone ou de brancher un câble réseau à son ordinateur pour communiquer est déjà désuet.

Être patient, prendre le temps d'apprendre ou de réfléchir deviendrait superflu : on veut tout, tout de suite, on veut *la* solution. Mais cela implique de confier de nombreuses décisions à de distants experts que l'on croit sur parole. Ce guide a pour but de proposer d'autres solutions, qui nécessitent de prendre le temps de les comprendre et de les appliquer.

Adapter ses pratiques à l'usage qu'on a du monde numérique est donc nécessaire dès lors qu'on veut, ou qu'on doit, apporter une certaine attention à son impact. Mais la traversée n'a que peu de sens en solitaire. Nous vous enjoignons donc à construire autour de vous votre radeau numérique, à sauter joyeusement à bord, sans oublier d'emmener ce guide et quelques fusées de détresse pour envoyer vos remarques à `guide@boum.org` (avec les précautions nécessaires).

Un « guide »

Ce guide est une tentative de rassembler ce que nous avons pu apprendre au cours de nos années de pratiques, d'erreurs, de réflexions et de discussions pour le partager.

1. On utilise ici le terme « flics » tel qu'il est défini dans l'introduction de *Face à la police / Face à la justice* [http://guidejuridique.net/].

Non seulement les technologies évoluent très vite, mais nous avons pu commettre des erreurs ou écrire des contre-vérités dans ces pages. Nous tenterons donc de tenir ces notes à jour à l'adresse : https://guide.boum.org/

Afin de rendre le tout plus digeste, nous avons divisé tout ce que nous souhaitions raconter en plusieurs tomes. Qu'on se trouve avec uniquement un ordinateur, que ce dernier soit connecté à un réseau ou qu'on soit chez soi ou au téléphone, cela représente des contextes différents, donc des menaces, des envies et des réponses différentes elles aussi.

Préface à la troisième édition

Il y a de ça trois ans, nous répétions déjà que les technologies évoluaient vite. Force est de constater qu'on n'arrête pas le progrès, et cela appelait bien sûr une troisième édition du *Guide*.

Entre autres scandales, celui des documents confidentiels de la *National Security Agency*[2], fuités par Edward Snowden[3], a permis de prendre la mesure de la panoplie de logiciels malveillants, de matériel espion, et autres systèmes de surveillance, déployés à l'échelle du monde. Les scénarios les plus alarmistes sont finalement le tissu du quotidien en matière de surveillance électronique. Malgré la propagation d'un sentiment d'impuissance, ces différentes révélations sur l'état général de la surveillance numérique rendent d'autant plus nécessaire de se donner les moyen d'y faire face.

L'année 2013 a vu la parution d'études concernant la robustesse toute relative des mots de passe, études qui permettent de mettre fin à d'anciennes intuitions et de bons conseils qui n'en sont plus.

Plusieurs nouveaux avertissements se sont donc glissés ça et là au fil des prochaines pages.

Sur le plan légal, la Loi relative à la Programmation Militaire a été inscrite dans la loi française fin 2013. Cet arsenal élargit les pretextes légaux permettant de surveiller les internautes, le nombre de ministères dont ces requêtes de surveillance peuvent émaner, ainsi que le nombre d'entités auxquelles ces requêtes d'adressent. Elle ouvre également la voie à la collecte *en temps réel* de ces données.

Du côté des outils, le printemps 2013 a vu la sortie de la nouvelle version de Debian, baptisée « *Wheezy* ». En juillet 2014 sortait la version 1.1 du système *live Tails*, dorénavant basé sur Wheezy, et possédant de nouvelles fonctionnalités majeures. Il a donc fallu revoir les outils pour que les recettes fonctionnent sur ces nouveaux systèmes.

La persistance est une de ces nouvelles fonctionnalités. Il est dorénavant possible de conserver, sur une partition chiffrée, des données ainsi que des configurations d'une session à l'autre. L'amnésie n'est plus totale !

La suppression de métadonnées contenues dans de nombreux types de fichiers est désormais simplifiée grâce au logiciel *Metadata Anonymisation Toolkit*, ainsi que la suppression sécurisée de fichiers, tout cela étant maintenant intégré dans Debian.

Enfin le calcul des sommes de contrôles est également simplifié par la mise en place d'un outil graphique permettant de les effectuer.

2. Wikipédia, 2014, *NSA* [https://fr.wikipedia.org/wiki/NSA]
3. Wikipédia, 2014, *Edward Snowden* [https://fr.wikipedia.org/wiki/Edward_Snowden]

Et pour les personnes ayant déjà un système installé avec la version précédente de Debian (Squeeze), un nouveau chapitre de cette troisième édition explique comment procéder à la mise à jour vers Debian Wheezy.

Grâce à cette révision, nous espérons que les pages suivantes restent d'une compagnie avisée dans la traversée de la jungle numérique… du moins, jusqu'à la suivante.

Tome 1

Hors connexions

Sommaire

Préface **iii**
Les revers de la mémoire numérique . . . iii
Rien à cacher ? . . . iii
Comprendre pour pouvoir choisir . . . iv
Prendre le temps de comprendre . . . iv
Un « guide » . . . iv

Préface à la troisième édition **vii**

Sommaire **1**

I Comprendre **7**

1 Quelques bases sur les ordinateurs **9**
1.1 Des machines à traiter les données . . . 9
1.2 Le matériel . . . 9
1.3 Électricité, champs magnétiques et ondes radios . . . 14
1.4 Les logiciels . . . 14
1.5 Le rangement des données . . . 16

2 Traces à tous les étages **19**
2.1 Dans la mémoire vive . . . 19
2.2 Dans la mémoire virtuelle . . . 19
2.3 Veille et hibernation . . . 20
2.4 Les journaux . . . 20
2.5 Sauvegardes automatiques et autres listes . . . 21
2.6 Les méta-données . . . 21

3 Malware, mouchards et autres espions **23**
3.1 Les logiciels malveillants . . . 24
3.2 Les matériel espions . . . 26
3.3 Les *keyloggers*, ou enregistreurs de frappe au clavier . . . 26
3.4 Des problèmes d'impression ? . . . 27

4 Quelques illusions de sécurité... **29**
4.1 Logiciels propriétaires, *open source*, libres . . . 29
4.2 Le mot de passe d'un compte ne protège pas ses données . . . 31
4.3 À propos de l'« effacement » des fichiers . . . 32
4.4 Les logiciels portables : une fausse solution . . . 34

5 La cryptographie **37**
5.1 Protéger des données des regards indiscrets . . . 37
5.2 S'assurer de l'intégrité de données . . . 41
5.3 Symétrique, asymétrique ? . . . 43

II Choisir des réponses adaptées **47**

6 Évaluation des risques **49**
6.1 Que veut-on protéger ? . . . 49
6.2 Contre qui veut-on se protéger ? . . . 49

7 Définir une politique de sécurité **51**
7.1 Une affaire de compromis . . . 51
7.2 Comment faire ? . . . 52
7.3 Quelques règles . . . 52

8 Un nouveau départ **57**
8.1 Contexte . . . 57
8.2 Évaluer les risques . . . 57
8.3 Définir une politique de sécurité . . . 58

9 Travailler sur un document sensible **65**
9.1 Contexte . . . 65
9.2 Évaluer les risques . . . 65
9.3 Accro à Windows ? . . . 66
9.4 Le système *live* amnésique . . . 66
9.5 Travailler sur un document sensible… sur un système *live* . . . 68
9.6 Travailler sur un document sensible… sous Windows . . . 68
9.7 Nettoyer les métadonnées du document terminé . . . 75
9.8 Limites communes à ces politiques de sécurité . . . 75

10 Archiver un projet achevé **77**
10.1 Contexte . . . 77
10.2 Est-ce bien nécessaire ? . . . 77
10.3 Évaluer les risques . . . 77
10.4 Méthode . . . 78
10.5 Quelle phrase de passe ? . . . 78
10.6 Un disque dur ? Une clé ? Plusieurs clés ? . . . 79

III Outils **81**

11 Utiliser un terminal **85**
11.1 Qu'est-ce qu'un terminal ? . . . 85
11.2 À propos des commandes . . . 86
11.3 Terminal ? Terminal administrateur ? . . . 88
11.4 Encore une mise en garde . . . 88
11.5 Un exercice . . . 88
11.6 Attention aux traces ! . . . 90
11.7 Pour aller plus loin . . . 90

12 Choisir une phrase de passe **91**

13 Démarrer sur un CD, un DVD ou une clé USB **93**
13.1 Essayer naïvement . . . 93
13.2 Tenter de choisir le périphérique de démarrage . . . 93
13.3 Modifier les paramètres du micrologiciel . . . 94

14 Utiliser un système *live* **99**
14.1 Des systèmes *live* discrets . . . 99
14.2 Télécharger et installer Tails . . . 100
14.3 Cloner ou mettre à jour une clé Tails . . . 101
14.4 Démarrer sur un système *live* . . . 101
14.5 Utiliser la persistance de Tails . . . 102

15 Installer un système chiffré **105**
15.1 Limites . . . 105
15.2 Télécharger un support d'installation . . . 106
15.3 Vérifier l'empreinte du support d'installation . . . 107
15.4 Préparer les supports d'installation . . . 107
15.5 L'installation proprement dite . . . 110
15.6 Quelques pistes pour continuer . . . 114
15.7 Un peu de documentation sur Debian et GNU/Linux . . . 114

16 Choisir, vérifier et installer un logiciel **115**
16.1 Trouver un logiciel . . . 116
16.2 Critères de choix . . . 117
16.3 Installer un paquet Debian . . . 120
16.4 Comment modifier ses dépôts Debian . . . 122

17 Effacer des données « pour de vrai » **127**
17.1 Un peu de théorie . . . 127
17.2 Sur d'autres systèmes . . . 128
17.3 Allons-y . . . 128
17.4 Supprimer des fichiers… et leur contenu . . . 129
17.5 Effacer « pour de vrai » tout un disque . . . 129
17.6 Effacer tout le contenu d'un disque . . . 130
17.7 Effacer le contenu d'une partition chiffrée LUKS . . . 132
17.8 Rendre irrécupérables des données déjà supprimées . . . 135

18 Partitionner et chiffrer un disque dur **137**
18.1 Chiffrer un disque dur avec LUKS et `dm-crypt` . . . 137
18.2 D'autres logiciels que l'on déconseille . . . 138
18.3 En pratique . . . 138
18.4 Préparer un disque à chiffrer . . . 139
18.5 Créer une partition non chiffrée . . . 140
18.6 Créer une partition chiffrée . . . 140
18.7 Utiliser un disque dur chiffré . . . 141

19 Sauvegarder des données **143**
19.1 Gestionnaire de fichiers et stockage chiffré . . . 143
19.2 En utilisant Déjà Dup . . . 145

20 Partager un secret **149**
20.1 Partager une phrase de passe . . . 149
20.2 Reconstituer la phrase de passe . . . 151

21 Utiliser les sommes de contrôle **153**
21.1 Obtenir la somme de contrôle d'un fichier . . . 153
21.2 Vérifier l'intégrité d'un fichier . . . 154

22 Installer et utiliser un système virtualisé **155**
22.1 Installer VirtualBox . . . 156
22.2 Installer un Windows virtualisé . . . 157
22.3 Sauvegarder une image de disque virtuel propre . . . 160
22.4 Effacer « pour de vrai » une machine virtuelle . . . 161
22.5 Créer une nouvelle machine virtuelle à partir d'une image propre . . . 162
22.6 Envoyer des fichiers à un système virtualisé . . . 164
22.7 Faire sortir des fichiers d'un système virtualisé . . . 166

23 Garder un système à jour **169**
23.1 Garder Tails à jour . . . 169
23.2 Garder à jour un système chiffré . . . 169
23.3 Les mises à jour quotidiennes d'un système chiffré . . . 170
23.4 Passage à une nouvelle version stable . . . 170

24 Nettoyer les métadonnées d'un document **175**
24.1 Installer les logiciels nécessaires . . . 175
24.2 Ouvrir le *Metadata Anonymisation Toolkit* . . . 175
24.3 Ajouter des fichiers à nettoyer . . . 175
24.4 Nettoyer les fichiers . . . 176

Qui parle ? **177**

Index **179**

Première partie

Comprendre

Devant la grande complexité des outils informatiques et numériques, la quantité d'informations à avaler pour tenter d'acquérir quelques pratiques d'autodéfense peut paraître énorme. Elle l'est sûrement pour qui chercherait à tout comprendre en même temps…

Ce premier tome se concentrera donc sur l'utilisation d'un ordinateur « hors connexion » — on pourrait aussi bien dire *préalablement à toute connexion*. Mais ce sont aussi des connaissances plus générales qui valent *que l'ordinateur soit connecté ou non* à un réseau. On met donc de côté, jusqu'au second tome, les menaces spécifiquement liées à l'usage d'Internet et des réseaux.

Pour ce morceau *hors connexion*, comme pour les autres, on prendra le temps de s'attarder sur des notions de base, leurs implications en termes de sécurité / confidentialité / intimité [1]. Après l'analyse de cas concrets d'utilisation, on pourra se pencher sur quelques recettes pratiques.

Une dernière précision avant de nous jeter à l'eau : *l'illusion de sécurité est bien pire que la conscience nette d'une faiblesse*. Aussi, prenons le temps de bien lire les premières parties avant de nous jeter sur nos claviers… ou même de jeter nos ordinateurs par les fenêtres.

1. On souhaite ici faire appel à une notion un peu floue : quelque chose qui tournerait autour de la possibilité de décider ce qu'on révèle, à qui on le révèle, ainsi que ce que l'on garde secret ; quelque chose qui inclurait aussi une certaine attention à déjouer les tentatives de percer ces secrets. Le terme employé en anglais pour nommer ce qu'on évoque ici est *privacy*. Aucun mot français ne nous semble adapté pour porter tout le sens que l'on aimerait mettre derrière cette notion. Ailleurs, on rencontrera souvent le terme « sécurité », mais l'usage qui en est couramment fait nous donne envie de l'éviter.

Chapitre 1

Quelques bases sur les ordinateurs

Commençons par le commencement.

Un *ordinateur*, ce n'est pas un chapeau de magicien où on peut ranger des lapins et les ressortir quand on a besoin, et qui permettrait en appuyant sur le bon bouton d'avoir une fenêtre ouverte sur l'autre bout du monde.

Un ordinateur est composé d'un ensemble de machines plus ou moins complexes, reliées entre elles par des connexions électriques, des câbles, et parfois des ondes radios. Tout ce *matériel* stocke, transforme et réplique des signaux pour manipuler l'information que l'on peut voir sur un bel écran avec plein de boutons où cliquer.

Comprendre comment s'articulent ces principaux composants, comprendre les bases de ce qui fait fonctionner tout ça, c'est la première étape pour comprendre où sont les forces et les faiblesses de ces engins, à qui l'on confie pas mal de nos données.

1.1 Des machines à traiter les données

Les ordinateurs sont des machines inventées pour pouvoir s'occuper d'informations. Elles savent donc précisemment enregistrer, traiter, analyser et classer de l'information, même en très grande quantité.

Dans le monde numérique, copier une information ne coûte que quelques micro-watts, autant dire pas grand'chose : c'est essentiel d'avoir ça en tête si nous voulons limiter l'accès à des informations.

Il faut tout simplement considérer que *mettre une information sur un ordinateur* (et c'est encore plus vrai quand il est sur un réseau), *c'est accepter que cette information puisse nous échapper.*

Ce guide peut aider à limiter la casse, mais il faut malgré tout prendre acte de cette réalité.

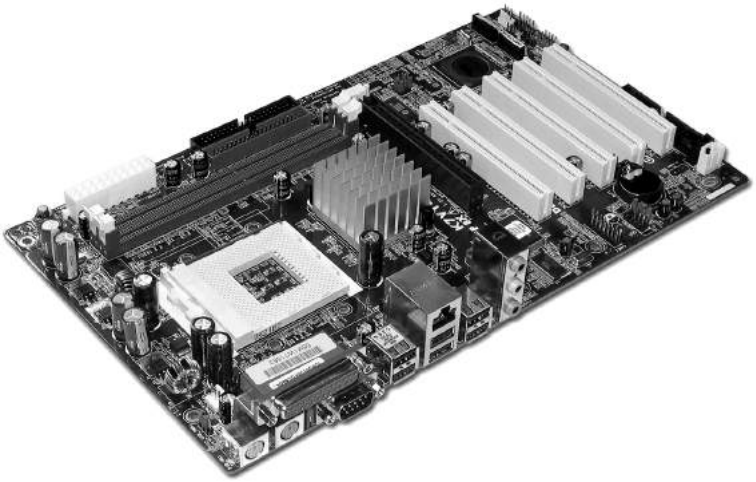
1.2 Le matériel

Somme de composants reliées entre eux, notre ordinateur est donc d'abord une accumulation d'objets, qu'on peut toucher, déplacer, bidouiller, casser.

L'ensemble *écran* / *clavier* / *tour* (ou unité centrale), ou l'ordinateur portable, est pratique quand on veut simplement brancher les fils aux bons endroits. Mais pour savoir ce qu'il advient de nos données, un examen plus fin est nécessaire.

On considère ici le contenu d'un ordinateur « classique », parfois appelé PC. Mais on retrouvera la plupart de ces composants avec de légères variations sur d'autres machines : Mac, téléphone portable, « box » de connexion à Internet, tablette, lecteur MP3, *etc.*

1.2.1 La carte-mère



Une carte-mère

Un ordinateur est surtout composé d'éléments électroniques. La *carte-mère* est un gros circuit imprimé qui permet de relier la plupart de ces éléments à travers l'équivalent de fils électriques. Sur la carte-mère viendront se brancher au minimum un processeur, de la mémoire vive, un système de stockage (disque dur ou autre mémoire), de quoi démarrer l'ordinateur (un micrologiciel) et d'autres cartes et périphériques selon les besoins.

On va rapidement faire un petit tour à travers tout ça pour avoir une vague idée de qui fait quoi, ce sera fort utile par la suite.

1.2.2 Le processeur

La puce d'un microprocesseur Intel Pentium 60 Mhz dans son boîtier

Le processeur (aussi appelé CPU, pour *central processing unit* ou « unité centrale de traitement » en français) est le composant qui s'occupe du traitement des données.

Pour se représenter le travail d'un processeur, l'exemple le plus concret sur lequel se baser est la calculatrice. Sur une calculatrice on entre des données (les nombres) et des opérations à faire dessus (addition, multiplication ou autres) avant d'examiner le résultat, éventuellement pour s'en servir ensuite comme base pour d'autres calculs.

Un processeur fonctionne exactement de la même manière. À partir de données (qui peuvent être la liste d'opérations à effectuer), il se contente d'exécuter à la chaîne les traitements à faire. Il ne fait que ça, mais il le fait vraiment très vite.

Mais si le processeur n'est qu'une simple calculatrice, comment peut-on alors effectuer des traitements sur des informations qui ne sont pas des nombres, par exemple sur du texte, des images, du son ou un déplacement de la souris ?

Tout simplement en transformant en nombre tout ce qui ne l'est pas, en utilisant un code défini auparavant. Pour du texte, ça peut par exemple être `A = 65`, `B = 66`, *etc.* Une fois ce code défini, on peut *numériser* notre information. Avec le code précédent, on peut par exemple transformer « GUIDE » en `71, 85, 73, 44, 69`.

Cette série de chiffres permet de représenter les lettres qui composent notre mot. Mais le processus de numérisation perdra toujours de l'information. Pour cet exemple, on perd au passage la spécificité de l'écriture manuscrite alors que pourtant, une rature, des lettres hésitantes constituent tout autant de « l'information ». Lorsque des choses passent dans le tamis du monde numérique, on perd forcément toujours des morceaux au passage.

Au-delà des données, les opérations que le processeur doit effectuer (ses *instructions*) sont également codées sous forme de nombres binaires. Un programme est donc une série d'instructions, manipulées comme n'importe quelles autres données.

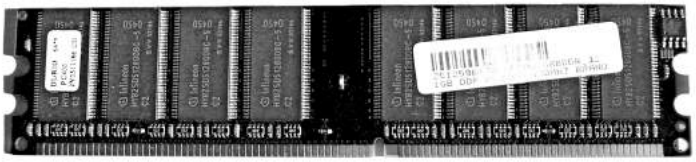
> À l'intérieur de l'ordinateur, tous ces nombres sont eux-mêmes représentés à l'aide d'états électriques : absence de courant, ou présence de courant. Il y a donc deux possibilités, ces fameux `0` et `1` que l'on peut croiser un peu partout. C'est pourquoi on parle de *bi*-naire. Et c'est uniquement à l'aide d'un paquet de fils et de plusieurs milliards de *transistors* (des interrupteurs, pas si différents de ceux pour allumer ou éteindre la lumière dans une cuisine) que le traitement des données se fait.

Tous les processeurs ne fonctionnent pas de la même manière. Certains ont été conçus pour être plus efficaces pour certains types de calcul, d'autres pour consommer le moins d'énergie, *etc.* Par ailleurs, tous les processeurs ne disposent pas exactement des mêmes instructions. Il en existe de grandes familles, que l'on appelle des *architectures*. Cela a son importance, car un programme prévu pour fonctionner sur une architecture donnée ne fonctionnera en général pas sur une autre.

1.2.3 La mémoire vive

La mémoire vive (ou RAM, pour *Random Access Memory*) se présente souvent sous forme de *barrettes*, et se branche directement sur la carte-mère.

La mémoire vive sert à stocker tous les logiciels et les documents ouverts. C'est à cet endroit que le processeur va chercher les données à traiter et entreposer le résultat des opérations. Ces informations doivent donc forcément s'y trouver sous une forme directement utilisable pour effectuer les calculs.



Une barrette de mémoire vive

L'accès à la mémoire vive est très rapide : il suffit du temps nécessaire pour basculer les interrupteurs qui vont relier le processeur à la case de la mémoire à lire (ou à écrire).

Lorsque la mémoire vive n'est plus alimentée en électricité, les données qu'elle contient deviennent illisibles après quelques minutes ou quelques heures, selon les modèles.

1.2.4 Le disque dur

Un disque dur 3 pouces ½

Étant donné que la mémoire vive s'efface à partir du moment où elle n'a plus de courant, l'ordinateur a besoin d'un autre endroit où stocker données et programmes entre chaque allumage. On parle aussi de mémoire *persistante* ou de mémoire *morte* : une mémoire où les informations écrites restent, même sans alimentation électrique.

Pour ce faire, on utilise en général un *disque dur*. C'est souvent une coque en métal dans laquelle se trouvent plusieurs disques qui tournent sans s'arrêter. Sur ces disques se trouvent de minuscules morceaux de fer. Au-dessus de chaque disque se trouvent des *têtes de lecture*. À l'aide de champs magnétiques, ces dernières détectent et modifient la position des morceaux de fer. C'est la position des morceaux de fer qui permet de coder les information à stocker.

Ce mécanisme est *beaucoup plus lent* — 50 fois environ — que l'accès à la mémoire vive. Par contre, c'est plus simple d'y mettre *beaucoup plus d'informations*.

Les informations que l'on met donc généralement sur un disque dur sont, bien entendu, des documents, mais aussi les programmes et toutes les données qu'ils utilisent pour

fonctionner, comme des fichiers temporaires, des journaux de bord, des fichiers de sauvegarde, des fichiers de configuration, *etc.*

Le disque dur conserve donc une mémoire quasi-permanente et quasi-exhaustive pour toutes sortes de traces qui parlent de nous, de ce que nous faisons, avec qui et comment, dès qu'on utilise un ordinateur.

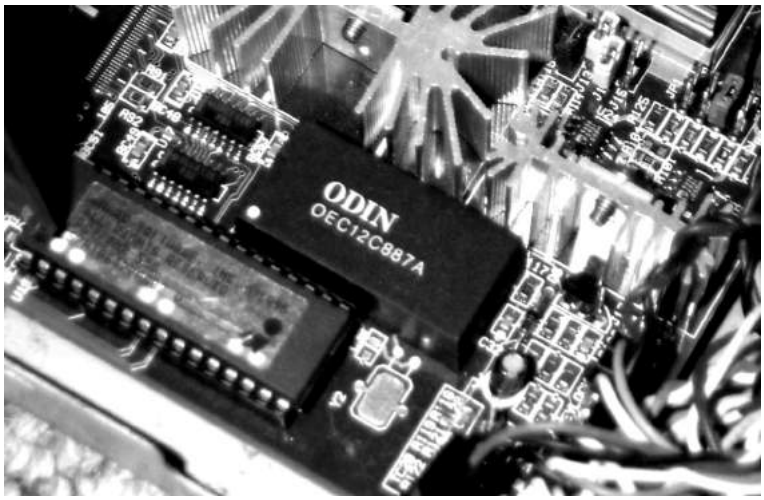
1.2.5 Les autres périphériques

Avec uniquement un processeur, de la mémoire vive et un support de stockage, on obtient déjà un ordinateur. Pas très causant, par contre. Donc on lui adjoint généralement d'autres *périphériques* comme un clavier, une souris, un écran, un adaptateur réseau (avec ou sans fil), un lecteur de DVD, *etc.*

Certains périphériques nécessitent des puces supplémentaires afin que le processeur puisse y accéder. Ces puces peuvent êtres soudées directement au circuit de la carte-mère (c'est typiquement le cas pour le clavier) ou alors nécessiter l'ajout d'un circuit supplémentaire, livré sous forme de carte (dite *fille*).

Afin de réduire le nombre de puces spécifiques (et donc coûteuses et compliquées à mettre au point), les systèmes d'accès aux périphériques tendent à s'uniformiser. Par exemple, la norme USB (pour *Universal Serial Bus*) est de plus en plus utilisée pour connecter imprimantes, claviers, souris, disques durs supplémentaires, adaptateurs réseaux ou ce qu'on appelle couramment des « clés USB ».

1.2.6 Le micrologiciel



Une puce de micrologiciel Award BIOS sur une carte-mère

Pour démarrer l'ordinateur, il faut donner au processeur un premier programme, pour pouvoir charger les programmes à exécuter ensuite.

Ce petit logiciel, appelé micrologiciel (*firmware* en anglais) est contenu dans une puce mémoire sur la carte mère. Cette mémoire fait partie d'un troisième type : la mémoire *flash*. C'est une mémoire qui garde les informations lorsqu'elle est éteinte, mais dont on ne peut remplacer le contenu que lors d'une opération qu'on appelle *flashage*. C'est aussi ce type de mémoire qu'on trouve dans les « clés USB » ou les « disques durs » dits *Solid State Disk* (ou SSD).

Le micrologiciel historique de la plupart des ordinateurs personnels était appelé BIOS (*Basic Input/Output System*, ou système d'entrée/sortie de base). Depuis 2012, de plus en plus d'ordinateurs utilisent un nouveau standard appelé UEFI (*Unified Extended Firmware Interface*).

page ci-contre

Ce premier programme qu'exécute l'ordinateur permet, entre autres, de choisir où se trouve le système d'exploitation que l'on veut utiliser (qui sera chargé à partir d'un disque dur, d'une clé USB, d'un CD ou d'un DVD, voire à partir du réseau).

1.3 Électricité, champs magnétiques et ondes radios

En ce qui concerne la confidentialité des informations qui circulent au sein d'un ordinateur, il faut déjà prendre acte de plusieurs choses après ce rapide tour de ce qui le compose.

Tout d'abord, l'essentiel de l'information circule sous forme de courants électriques. Rien n'empêche donc de mettre l'équivalent d'un bête *voltmètre* pour mesurer le courant qui passe, et ainsi pouvoir reconstituer n'importe quelles données manipulées par l'ordinateur sous une forme ou une autre.

Par ailleurs, tout courant qui circule a tendance à émettre un champ magnétique. Ces champs magnétiques peuvent rayonner à quelques mètres, voire plus [1]. Il est donc possible pour qui s'en donne les moyens de reconstituer le contenu d'un écran ou ce qui a été tapé sur un clavier, et cela, même derrière un mur, depuis la rue ou l'appartement contigu : ainsi, des chercheurs ont réussi à enregistrer les touches tapées sur des claviers filaires normaux à partir de leurs émissions électromagnétiques, à une distance allant jusqu'à 20 mètres [2].

Le même type d'opération est possible à partir de l'observation des légères perturbations [3] que génère l'ordinateur sur le réseau électrique où il est branché. Il faut toutefois pour cela que l'attaquant soit branché sur le même réseau électrique.

Enfin, certains périphériques (claviers, souris, écouteurs, *etc.*) fonctionnent *sans fil*. Ils communiquent alors avec l'ordinateur par des ondes radio que n'importe qui autour peut capter et éventuellement décoder sans vergogne.

Bref, pour résumer, même si un ordinateur n'est pas connecté à un réseau, et quels que soient les programmes qui fonctionnent, il reste tout de même possible pour des personnes bien équipées de réaliser une « écoute » de ce qui se passe à l'intérieur de l'ordinateur.

1.4 Les logiciels

Au-delà de la somme d'éléments *physiques* qui constituent un ordinateur, il faut aussi se pencher sur les éléments moins palpables : les logiciels.

À l'époque des tout premiers ordinateurs, chaque fois qu'il fallait exécuter des traitements différents, il fallait intervenir physiquement pour changer la disposition des câbles et des composants. On en est bien loin aujourd'hui : les opérations à réaliser pour faire les traitements sont devenues des données comme les autres. Des données qu'on appelle « programmes » qui sont chargées, modifiées, manipulées par d'autres programmes.

Ceux-ci sont généralement écrits pour essayer de ne faire qu'une seule chose, et de la faire bien, principalement dans le but de rester compréhensibles par les êtres humains qui les conçoivent. C'est ensuite l'interaction de dizaines de milliers de programmes

1. Berke Durak a réussi en 1995 à capter les ondes électromagnétiques [http://lambda-diode.com/electronics/tempest/] émises par la plupart des composants de son ordinateur avec un simple *walkman* capable de recevoir la radio.
2. Martin Vuagnoux et Sylvain Pasini ont réalisé d'effrayantes vidéos [http://lasecwww.epfl.ch/keyboard/] pour illustrer leur papier *Compromising Electromagnetic Emanations of Wired and Wireless Keyboards* publié en 2009.
3. Paul Kocher, Joshua Jaffe et Benjamin Jun ont publié en 1998, en anglais, un rapport [http://www.cryptography.com/public/pdf/DPATechInfo.pdf] expliquant les différentes techniques d'analyse de consommation électrique.

entre eux qui permettra de réaliser les tâches complexes pour lesquelles sont généralement utilisés les ordinateurs de nos jours.

L'effet produit lorsqu'on clique sur un bouton, c'est donc le lancement d'une chaîne d'évènements, d'une somme impressionnante de calculs, qui aboutissent à des impulsions électriques venant à la fin modifier un objet physique (comme un DVD qu'on veut graver, un écran qui modifie ses LEDs pour afficher une nouvelle page, ou un disque dur qui active ou désactive des micro-interrupteurs pour créer la suite binaire de données qui constituera un *fichier*).

1.4.1 Le système d'exploitation

Le but d'un système d'exploitation est avant tout de permettre aux logiciels de se partager l'accès aux composants matériels de l'ordinateur. Son rôle est aussi de permettre aux différents logiciels de communiquer entre eux. Un système d'exploitation est par ailleurs généralement livré avec des logiciels, au minimum de quoi permettre de démarrer d'autres logiciels.

La partie la plus fondamentale d'un système d'exploitation est son noyau, qui s'occupe de coordonner l'utilisation du matériel par les programmes.

Pour chaque composant matériel de l'ordinateur que l'on veut utiliser, le noyau active un programme qu'on appelle « pilote » (ou *driver* en anglais). Il existe des pilotes pour les périphériques d'entrée (comme le clavier et la souris), de sortie (écran, imprimante, *etc.*), de stockage (DVD, clé USB, *etc.*).

Le noyau gère aussi l'exécution des programmes, en leur donnant des morceaux de mémoire et en répartissant le temps de calcul du processeur entre les différents programmes qui veulent le faire travailler.

Au-delà du noyau, les systèmes d'exploitation utilisés de nos jours, comme Windows, Mac OS X ou GNU/Linux (avec Debian, Ubuntu, Fedora, par exemple) incluent aussi de nombreux utilitaires ainsi que des environnements de bureaux graphiques qui permettent d'utiliser l'ordinateur en cliquant simplement sur des boutons.

Le système d'exploitation est en général stocké sur le disque dur. Cependant, il est aussi tout à fait possible d'utiliser un système d'exploitation enregistré sur une clé USB ou gravé sur un DVD. Dans ce dernier cas, on parle de système *live* (vu qu'aucune modification ne pourra être faite sur le DVD).

1.4.2 Les applications

On appelle « applications » les logiciels qui permettent réellement de faire ce qu'on a envie de demander à l'ordinateur. On peut citer comme exemple Mozilla Firefox comme navigateur web, LibreOffice pour la bureautique ou encore GIMP ou Adobe Photoshop pour le traitement d'images.

Chaque système d'exploitation définit une méthode bien spécifique pour que les applications puissent accéder au matériel, à des données, au réseau, ou à d'autres ressources. Les applications que l'on souhaite utiliser doivent donc êtres conçues pour le système d'exploitation de l'ordinateur sur lequel on veut s'en servir.

1.4.3 Les bibliothèques

Plutôt que de réécrire dans toutes les applications des morceaux de programme chargés de faire les mêmes choses, les logiciels se les partagent dans des bibliothèques, ou *libraries* en anglais.

Il existe des bibliothèques pour l'affichage graphique (assurant une cohérence de ce qui est affiché à l'écran), pour lire ou écrire des formats de fichiers, pour interroger certains services réseaux, *etc.*

Si l'on n'est pas programmeur, on a rarement besoin de toucher aux bibliothèques. Il peut toutefois être intéressant de connaître leur existence, ne serait-ce que parce qu'un problème (comme une erreur de programmation) dans une bibliothèque peut se répercuter sur tous les logiciels qui l'utilisent.

1.5 Le rangement des données

On a vu qu'un disque dur (ou une clé USB) permettait de garder des données entre deux allumages d'un ordinateur.

Mais, histoire de s'y retrouver, les données sont agencées d'une certaine manière : un meuble dans lequel on aurait simplement entassé des feuilles de papier ne constitue pas vraiment une forme de rangement des plus efficaces…

1.5.1 Les partitions

Tout comme dans un meuble on peut mettre plusieurs étagères, on peut « découper » un disque dur en plusieurs *partitions*.

Chaque étagère pourra avoir une hauteur différente, un classement différent, selon que l'on souhaite y mettre des livres ou des classeurs, par ordre alphabétique ou par ordre de lecture.

De la même manière, sur un disque dur, chaque partition pourra être de taille différente et contenir un mode d'organisation différent : un système de fichiers.

1.5.2 Les systèmes de fichiers

Un système de fichiers sert avant tout à pouvoir retrouver des informations dans notre immense pile de données, comme la table des matières d'un livre de cuisine permet directement d'aller à la bonne page pour lire la recette du festin du soir.

Il peut être important de noter que la suppression d'un fichier ne fait qu'enlever une ligne dans la table des matières. En parcourant toutes les pages, on pourra toujours retrouver notre recette, tant que la page n'aura pas été réécrite — on développera cela plus tard.

page 32

On peut imaginer des milliers de formats différents pour ranger des données, et il existe donc de nombreux systèmes de fichiers différents. On parle de *formatage* lors de la création d'un système de fichiers sur un support.

Vu que c'est le système d'exploitation qui donne aux programmes l'accès aux données, un système de fichiers est souvent fortement lié à un système d'exploitation particulier.

> Pour en citer quelques-un : les type NTFS, FAT32 sont ceux employés habituellement par les systèmes d'exploitation Windows ; le type *ext* (`ext3`, `ext4`) est souvent utilisé sous GNU/Linux ; les types HFS, HFS+ et HFSX sont employés par Mac OS X.

Il est néanmoins possible de lire un système de fichiers « étranger » au système qu'on utilise, moyennant l'usage du logiciel adéquat. Windows est par exemple capable de lire une partition *ext3*, si on installe le logiciel approprié.

Une des conséquences de cela, c'est qu'il peut exister sur un ordinateur donné des espaces de stockage invisibles pour l'utilisateur parce que non reconnus par le système d'exploitation (ou non accessibles pour l'utilisateur), mais qui sont pourtant bel et bien présents.

1.5.3 Les formats de fichiers

Les données que l'on manipule sont généralement regroupées sous forme de fichiers. Un fichier a un contenu, mais aussi un nom, un emplacement (le dossier dans lequel il se trouve), une taille, et d'autres détails selon le système de fichiers utilisé.

Mais à l'intérieur de chaque fichier, les données sont elles-mêmes organisées différemment selon leur nature et les logiciels utilisés pour les manipuler. On parle de *format* de fichier pour les différencier.

En général, on met à la fin d'un fichier un code, qu'on appelle parfois *extension*, permettant d'indiquer le format du fichier.

Quelques exemples d'extensions : pour la musique, on utilisera le format MP3 ou Ogg, pour un document texte le LibreOffice ce sera OpenDocument Text (ODT), pour des images, on aura le choix entre le JPEG, le PNG, le format d'Adobe Photoshop (PSD), *etc.*

Il peut être intéressant de faire la différence entre les formats *ouverts*, dont les détails sont publics, et les formats *propriétaires*, souvent conçus pour être manipulés par un logiciel bien précis.

Les formats propriétaires ont parfois été observés à la loupe pour être ouverts par d'autres logiciels, mais leur compréhension reste souvent imparfaite et assujettie à des changements d'une version à l'autre d'une application. C'est typiquement le cas pour le format de Microsoft Word, souvent appelé `.doc` ou `.docx`.

1.5.4 La mémoire virtuelle (*swap*)

Normalement, toutes les données auxquelles le processeur doit accéder, et donc tous les programmes et les documents ouverts, devraient se trouver en mémoire vive. Mais pour pouvoir ouvrir plein de programmes et de documents, les systèmes d'exploitation modernes trichent : ils échangent, quand c'est nécessaire, des morceaux de mémoire vive avec un espace du disque dur dédié à cet effet. On parle alors de « mémoire virtuelle », de *swap* en anglais ou encore d'« espace d'échange ».

Le système d'exploitation fait donc sa petite cuisine pour que le processeur ait toujours dans la mémoire vive les données auxquelles il veut réellement accéder. La *swap* est ainsi un exemple d'espace de stockage auquel on ne pense pas forcément, enregistré sur le disque dur, soit sous forme d'un gros fichier contigu (sous Microsoft Windows), soit dans une partition à part (avec Linux).

On reviendra dans la partie suivante sur les problèmes que posent ces questions de format et d'espaces de stockage en termes de confidentialité des données.

Chapitre 2

Traces à tous les étages

Le fonctionnement normal d'un ordinateur laisse de nombreuses traces de ce que l'on fait dessus. Parfois, elles sont *nécessaires* à son fonctionnement. D'autres fois, ces informations sont collectées pour permettre aux logiciels d'être « plus pratiques ».

2.1 Dans la mémoire vive

On vient de voir que le premier lieu de stockage des informations sur l'ordinateur est la mémoire vive. page 11

Tant que l'ordinateur est sous tension électrique, elle contient toutes les informations dont le système a besoin. Elle conserve donc nécessairement de nombreuses traces : frappes au clavier (y compris les mots de passe), fichiers ouverts, évènements divers qui ont rythmé la phase d'éveil de l'ordinateur.

En prenant le contrôle d'un ordinateur qui est allumé, il n'est pas très difficile de lui faire cracher l'ensemble des informations contenues dans la mémoire vive, par exemple vers une clé USB ou vers un autre ordinateur à travers le réseau. Et prendre le contrôle d'un ordinateur peut être aussi simple qu'y brancher un *iPod* quand on a le dos tourné [1]. Une fois récupérées, les nombreuses informations que contient la mémoire vive sur l'ordinateur et les personnes qui l'utilisent pourront alors être exploitées…

Par ailleurs, si ces données deviennent illisibles lors de la mise hors tension, cela prend néanmoins du temps, ce qui peut suffire pour qu'une personne mal intentionnée ait le temps de récupérer ce qui s'y trouve. On appelle cela une « *cold boot attack* » : l'idée est de copier le contenu de la mémoire vive avant qu'elle ait eu le temps de s'effacer, de manière à l'exploiter par la suite. Il est même techniquement possible de porter à très basse température la mémoire d'un ordinateur fraîchement éteint — auquel cas on peut faire subsister son contenu plusieurs heures, voire plusieurs jours [2].

Cette attaque doit cependant être réalisée peu de temps après la mise hors tension. Par ailleurs, si on utilise quelques gros logiciels (par exemple en retouchant une énorme image avec Adobe Photoshop ou GIMP) avant d'éteindre son ordinateur, les traces qu'on a laissées précédemment en mémoire vive ont de fortes chances d'être recouvertes. Mais surtout, il existe des logiciels spécialement conçus pour écraser le contenu de la mémoire vive avec des données aléatoires.

2.2 Dans la mémoire virtuelle

Comme expliqué auparavant, le système d'exploitation utilise, dans certains cas, une page 17

1. Maximillian Dornseif, 2004, *0wned by an iPod* [https://pacsec.jp/psj04/psj04-dornseif-e.ppt]. Bruce Schneier, 2006, *Hacking Computers Over USB* [http://www.schneier.com/blog/archives/2006/06/hacking_compute.html] (en anglais).
2. J. Alex Halderman et Al., 2008, *Least We Remember : Cold Boot Attacks on Encryption Keys* [http://citp.princeton.edu/memory/] (en anglais).

partie du disque dur pour venir en aide à sa mémoire vive. Ça arrive en particulier si l'ordinateur est fortement sollicité, par exemple quand on travaille sur de grosses images, mais aussi dans de nombreux autres cas, de façon peu prévisible.

La conséquence la plus gênante de ce système pourtant bien pratique, c'est que l'ordinateur va écrire sur le disque dur des informations qui se trouvent dans la mémoire vive… informations potentiellement sensibles, donc, *et qui resteront lisibles après avoir éteint l'ordinateur*.

page 34

Avec un ordinateur configuré de façon standard, il est donc illusoire de croire qu'un document lu à partir d'une clé USB, même ouvert avec un logiciel portable, ne laissera jamais de traces sur le disque dur.

page 37

Pour éviter de laisser n'importe qui accéder à ces données, il est possible d'utiliser un système d'exploitation configuré pour chiffrer la mémoire virtuelle.

2.3 Veille et hibernation

La plupart des systèmes d'exploitation permettent de mettre un ordinateur « en pause ». C'est surtout utilisé avec les ordinateurs portables mais c'est également valable pour les ordinateurs de bureau.

Il y a deux grandes familles de « pause » : la veille et l'hibernation.

2.3.1 La veille

La *veille* (appelée aussi en anglais *suspend to ram* ou *suspend*) consiste à éteindre le maximum de composants de l'ordinateur tout en gardant sous tension de quoi pouvoir le rallumer rapidement.

Au minimum, la mémoire vive continuera d'être alimentée pour conserver l'intégralité des données sur lesquelles on travaillait — c'est-à-dire notamment les mots de passe et les clés de chiffrement.

page 37

Bref, un ordinateur en veille protège aussi peu l'accès aux données qu'un ordinateur allumé.

2.3.2 L'hibernation

L'*hibernation* ou *mise en veille prolongée*, appelée aussi en anglais *suspend to disk*, consiste à sauvegarder l'intégralité de la mémoire vive sur le disque dur pour ensuite éteindre complètement l'ordinateur. Lors de son prochain démarrage, le système d'exploitation détectera l'hibernation, re-copiera la sauvegarde vers la mémoire vive et recommençera à travailler à partir de là.

page 17

Sur les systèmes GNU/Linux, la copie de la mémoire se fait généralement dans la *swap*. Sur d'autres systèmes, ça peut être dans un gros fichier, souvent caché.

Vu que c'est le contenu de la mémoire vive qui est écrite sur le disque dur, ça veut dire que tous les programmes et documents ouverts, mots de passe, clés de chiffrement et autres, pourront être retrouvés par quiconque accèdera au disque dur. Et cela, aussi longtemps que rien n'aura été réécrit par-dessus.

page 37

Ce risque est toutefois limité par le chiffrement du disque dur : la phrase de passe sera alors nécessaire pour accéder à la sauvegarde de la mémoire vive.

2.4 Les journaux

Les systèmes d'exploitation ont une forte tendance à écrire dans leur journal de bord un historique détaillé de ce qu'ils fabriquent.

Ces journaux (aussi appelés *logs*) sont utiles au système d'exploitation pour fonctionner, et permettent de corriger des problèmes de configuration ou des *bugs*.

Cependant leur existence peut parfois être problématique. Les cas de figure existants sont nombreux, mais les quelques exemples suivants devraient être suffisants pour donner une idée de ce risque :

- sous GNU/Linux, le système garde la date, l'heure et le nom de l'utilisateur qui se connecte chaque fois qu'un ordinateur est allumé ;
- toujours sous GNU/Linux, la marque et le modèle de chaque support amovible (disque externe, clé USB…) branché sont habituellement conservés ;
- sous Mac OS X, la date d'une impression et le nombre de pages sont inscrits dans les journaux ;
- sous Windows, le *moniteur d'évènements* enregistre le nom du logiciel, la date et l'heure de l'installation ou de la désinstallation d'une application.

2.5 Sauvegardes automatiques et autres listes

En plus de ces journaux, il est possible que d'autres traces de fichiers, même supprimés, subsistent sur l'ordinateur. Même si les fichiers et leur contenu ont été bien supprimés, une partie du système d'exploitation ou d'un autre programme peut en garder une trace délibérée.

Voici quelques exemples :

- sous Windows, Microsoft Office peut garder la référence d'un nom de fichier déjà supprimé dans le menu des « documents récents », et parfois même garder des fichiers temporaires avec le contenu du fichier en question ;
- sous GNU/Linux, un fichier d'historique peut contenir le nom d'un fichier préalablement supprimé. Et LibreOffice peut garder autant de traces d'un fichier supprimé que Microsoft Office. En pratique, il existe des dizaines de programmes fonctionnant ainsi ;
- lorsqu'on utilise une imprimante, le système d'exploitation copie souvent le fichier en attente dans la « file d'impression ». Le contenu de ce fichier, une fois la file vidée, n'aura pas disparu du disque dur pour autant ;
- sous Windows, lorsqu'on connecte un lecteur amovible (clé USB, disque dur externe, CD ou DVD), le système commence souvent par explorer son contenu afin de proposer des logiciels adaptés à sa lecture : cette exploration automatique laisse en mémoire la liste de tous les fichiers présents sur le support employé, même si aucun des fichiers qu'il contient n'est consulté.

Il est difficile de trouver une solution adéquate à ce problème. Un fichier, même parfaitement supprimé, continuera probablement à exister sur l'ordinateur pendant un certain temps sous une forme différente. Une recherche sur les données brutes du disque permettrait de voir si des copies de ces données existent ou pas… sauf si elles y sont seulement référencées, ou stockées sous une forme différente ; sous forme compressée, par exemple.

En fait, seul l'écrasement de la totalité du disque et l'installation d'un nouveau système d'exploitation permettent d'avoir la garantie que les traces d'un fichier ont bien été supprimées. Et dans une autre perspective, l'utilisation d'un système *live*, dont l'équipe de développement porte une attention particulière à cette question, garantit que ces traces ne seront pas laissées ailleurs que dans la mémoire vive. Nous y reviendrons. page 127

2.6 Les méta-données

En plus des informations contenues dans un fichier, il existe des informations accompagnant celui-ci, qui ne sont pas forcément visibles de prime abord : date de création,

nom du logiciel, de l'ordinateur, *etc.* Ces « données sur les données » s'appellent communément des « méta-données ».

page 16 Une partie des méta-données est enregistrée par le système de fichiers : le nom du fichier, la date et l'heure de création et de modification, et souvent bien d'autres page 19 choses. Ces traces sont laissées sur l'ordinateur (ce qui peut déjà être un problème en soi), mais celles-ci ne sont la plupart du temps pas inscrites dans le fichier.

page 17 En revanche, de nombreux formats de fichiers conservent également des méta-données *à l'intérieur* du fichier. Elles seront donc diffusées lors d'une éventuelle copie sur une clé USB, ou lors de l'envoi d'un email ou d'une publication en ligne. Ces informations pourront être connues de quiconque aura accès au fichier.

Les méta-données enregistrées dépendent des formats et des logiciels utilisés. La plupart des fichiers audio permettent d'y enregistrer le titre du morceau et l'interprète. Les traitements de texte ou les PDFs enregistreront un nom d'auteur, la date et l'heure de création, et parfois même l'historique complet des dernières modifications [3], et donc, potentiellement, des informations que l'on pensait avoir supprimé…

La palme revient probablement aux formats d'images comme TIFF ou JPEG : ces fichiers de photos créés par un appareil numérique ou un téléphone portable contiennent des méta-données au format EXIF. Ce dernier peut contenir la marque, le modèle et le numéro de série de l'appareil utilisé, mais aussi la date, l'heure et parfois les coordonnées géographiques de la prise de vue, sans oublier une version miniature de l'image. Ce sont d'ailleurs ces méta-données qui mettront fin à la cavale John McAfee, fondateur et ancien patron de la société de sécurité informatique du même nom [4]. Et toutes ces informations ont tendance à rester après être passées par un logiciel de retouche photo. Le cas de la miniature est particulièrement intéressant : de nombreuses photos disponibles sur Internet contiennent encore l'intégralité d'une photo recadrée… et des visages ayant été « floutés ». [5]

page 175 Pour la plupart des formats de fichiers *ouverts*, il existe toutefois des logiciels pour examiner et éventuellement supprimer les méta-données.

3. Deblock Fabrice, 2006, *Quand les documents Word trahissent la confidentialité* [http://www.journaldunet.com/solutions/0603/060327-indiscretions-word.shtml].

4. Big Browser, 2012, *VICE DE FORME – La bourde qui a mené à l'arrestation de John McAfee* [http://bigbrowser.blog.lemonde.fr/2012/12/12/vice-de-forme-la-bourde-qui-a-mene-a-larrestation-de-john-mcafee/].

5. Maximillian Dornseif et Steven J. Murdoch, 2004, *Hidden Data in Internet Published Documents* [http://events.ccc.de/congress/2004/fahrplan/files/316-hidden-data-slides.pdf] (en anglais).

Chapitre 3

Logiciels malveillants, mouchards et autres espions

Au-delà des traces que le fonctionnement de tout système d'exploitation laisse au moins le temps où l'ordinateur fonctionne, on peut aussi trouver dans nos ordinateurs tout un tas de *mouchards*. Soit installés à notre insu (permettant par exemple de détourner les journaux vers d'autres fins), soit présents de manière systématique dans les logiciels qu'on aura installés.

page 20

Ces mouchards peuvent participer à diverses techniques de surveillance, de la « lutte » contre le « piratage » de logiciels propriétaires, au fichage ciblé d'un individu, en passant par la collecte de données pour des pourriels (*spam*) ou autres arnaques.

page 29

La portée de ces dispositifs augmente fortement dès que l'ordinateur est connecté à Internet. Leur installation est alors grandement facilitée si on ne fait rien de spécial pour se protéger, et la récupération des données collectées se fait à distance.

Toutefois les gens qui récoltent ces informations sont inégalement dangereux : ça dépend des cas, de leurs motivations et de leurs moyens. Les proches abusifs [1], les sites Internet à la recherche de consommateurs à cibler, les multinationales comme Microsoft, les gendarmes de Saint-Tropez, ou la *National Security Agency* américaine… autant de structures souvent en concurrence entre elles et ne formant pas une totalité cohérente.

Pour s'introduire dans nos ordinateurs, ils n'ont pas accès aux mêmes passe-partout, et ne savent pas tous manipuler le pied-de-biche aussi bien : par exemple, l'espionnage industriel est une des raisons importantes de la surveillance plus ou moins légale [2], et, malgré les apparences [3], il ne faut pas croire que Microsoft donne toutes les astuces de Windows à la police française.

1. Catherine Armitage, 2014, *Spyware's role in domestic violence* [http://www.theage.com.au/technology/technology-news/spywares-role-in-domestic-violence-20140321-358sj.html] parle de l'utilisation des malwares et autres outils technologiques par des proches abusifs ou violents (en anglais).

2. Pour se faire une idée des problématiques liées à l'espionnage industriel Wikipédia, 2014, *Espionnage industriel* [https://fr.wikipedia.org/wiki/Espionnage_industriel].

3. Microsoft, en partenariat avec Interpol, a fabriqué une boîte à outils appelée COFEE (Computer Online Forensic Evidence Extractor) mise à disposition des polices d'une quinzaine de pays.Korben, 2009, *Cofee – La clé sécurité de Microsoft vient d'apparaitre sur la toile* [http://korben.info/cofee-la-cle-securite-de-microsoft-vient-dapparaitre-sur-la-toile.html].

Cependant, les services de sécurité français disposent maintenant des moyens de mettre en place une surveillance informatique très complète en toute légalité, en s'appuyant sur plusieurs « mouchards » présentés par la suite, à travers la Loi d'Orientation et de Programmation pour la Performance de la Sécurité Intérieure (dite LOPPSI 2). Ce texte inclut en effet des dispositions légales inédites permettant, dans le cadre d'une enquête sur des infractions relevant de criminalité ou de la délinquance organisée, d'installer des mouchards pour enregistrer et transmettre ce qui s'affiche à l'écran ou ce qui est entré au clavier d'un ordinateur, sans nécessairement disposer d'un accès physique à la machine, ou en pénétrant dans le domicile de la personne surveillée pour y installer les outils nécessaires [4].

3.1 Les logiciels malveillants

Les logiciels malveillants [5] (que l'on appelle également *malwares*) sont des logiciels qui ont été développés dans le but de nuire : collecte d'informations, hébergement d'informations illégales, relai de pourriel *etc.* Les virus informatiques, les vers, les chevaux de Troie, les *spyware*, les *rootkits* (logiciels permettant de prendre le contrôle d'un ordinateur) et les *keyloggers* sont de cette engeance. Certains programmes peuvent appartenir à plusieurs de ces catégories simultanément.

page 26

Afin de s'installer sur un ordinateur, certains logiciels malveillants exploitent les vulnérabilités du système d'exploitation [6] ou des applications. Ils s'appuient sur des erreurs de conception ou de programmation pour détourner le déroulement des programmes à leur avantage. Malheureusement, de telles « failles de sécurité » ont été trouvées dans de très nombreux logiciels, et de nouvelles sont trouvées constamment, tant par des gens qui cherchent à les corriger que par d'autres qui cherchent à les exploiter.

Un autre moyen courant est d'inciter la personne utilisant l'ordinateur à lancer le logiciel malveillant en le cachant dans un logiciel en apparence inoffensif. C'est ainsi qu'un simple lien vers une vidéo posté sur un réseau social lié à la révolution syrienne amenait en fait les internautes à télécharger un virus contenant un keylogger [7]. L'attaquant n'est alors pas obligé de trouver des vulnérabilités sérieuses dans des logiciels courants. Il est particulièrement difficile de s'assurer que des ordinateurs partagés par de nombreuses personnes ou des ordinateurs qui se trouvent dans des lieux publics, comme une bibliothèque ou un cybercafé, n'ont pas été corrompus : il suffit en effet qu'une seule personne un peu moins vigilante se soit faite avoir…

4. Pour plus de détails, nous recommandons la lecture de ces deux articles ; Marc Rees, 2009, *LOPPSI : la police sera autorisée à installer des chevaux de Troie* [http://www.pcinpact.com/actu/news/51027-police-opj-cheval-troie-loppsi.htm] et Marc Rees, 2009, *Les chevaux de Troie de la police seront installables à distance* [http://www.pcinpact.com/actu/news/51077-loppsi-chevaux-troie-police-distance.htm], et éventuellement du texte de loi Légifrance, 2011, *Section 6 bis : De la captation des données informatiques* [http://www.legifrance.gouv.fr/affichCode.do?cidTexte=LEGITEXT000006071154&idSectionTA=LEGISCTA000023712495&dateTOexte=20110428].

5. Toute cette partie est grandement inspirée du passage consacré à la question dans le *Surveillance Self-Defense Guide* [https://ssd.eff.org/tech/malware] de l'*Electronic Frontier Foundation*.

6. D'après l'*Internet Storm Center* [http://isc.sans.org/survivaltime.html], une installation de Microsoft Windows sur laquelle les mises à jour de sécurité n'ont pas été faites se fait compromettre en moins de 4 minutes si elle est connectée directement à Internet.

7. Eva Galperin et Al., 2014, *Quantum of Surveillance : Familiar Actors and Possible False Flags in Syrian Malware Campaigns* [https://www.eff.org/files/2013/12/28/quantum_of_surveillance4d.pdf] (en anglais).

En outre, la plupart des logiciels malveillants « sérieux » ne laissent pas de signe immédiatement visible de leur présence, et peuvent même être très difficiles à détecter. Le cas sans doute le plus compliqué est celui de failles jusqu'alors inconnues, appelées « exploits 0 day » [8], et que les logiciels antivirus seraient bien en mal de reconnaître, car pas encore répertoriées. C'est exactement ce genre d'exploitation de failles « 0 day » que la compagnie VUPEN a vendu à la NSA en 2012 [9].

En 2006, Joanna Rutkowska a présenté lors de la conférence *Black Hat* le *malware* nommé « Blue Pill ». Cette démonstration a montré qu'il était possible d'écrire un *rootkit* utilisant les technologies de virtualisation pour tromper le système d'exploitation et rendre ainsi vraiment très difficile d'identifier la présence du *malware*, une fois celui-ci chargé.

Ces logiciels peuvent voler les mots de passe, lire les documents stockés sur l'ordinateur (même les documents chiffrés, s'ils ont été déchiffrés à un moment), réduire à néant des dispositifs d'anonymat sur Internet, prendre des captures d'écran du bureau et se cacher eux-mêmes des autres programmes. Mais ils peuvent également utiliser le micro, la webcam ou d'autres périphériques de l'ordinateur. Il existe un vrai marché spécialisé où l'on peut acheter de tels programmes, personnalisés pour différents objectifs.

page 37

Ces logiciels peuvent concurrir à de nombreuses tâches : obtenir des numéros de cartes bancaires, des mots de passe de compte *PayPal*, à envoyer des pourriels ou à participer à attaquer un serveur en le saturant de demandes. Mais ils servent tout aussi bien à espionner des organisations ou des individus spécifiques [10].

Pour donner un exemple venu des États-Unis, le FBI a écrit un logiciel nommé CIPAV pour *Computer and Internet Protocol Address Verifier*. Ce dernier a permis entre autres d'identifier un adolescent de quinze ans ayant envoyé par email des menaces d'attentat contre un lycée de Washington [11].

Plus récemment, la loi HADOPI exige des internautes qu'ils « sécurisent » leur connexion, sous peine d'être responsables des usages illicites qui en seraient faits. Pour cela, l'autorité chargée de l'application de la loi eut la bonne idée de proposer aux internautes d'installer *volontairement* un logiciel espion qui enregistrerait un tas de données concernant l'usage de leur connexion, ainsi que celles permettant d'identifier les machines qui l'ont utilisée [12].

Personne ne sait combien d'ordinateurs sont infectés par des logiciels malveillants, mais certains estiment que c'est le cas pour **40 à 90 %** des installations de Windows.

8. Wikipédia, 2014, *Vulnérabilité jour zéro* [https://fr.wikipedia.org/wiki/Vulnérabilité_zero-day].
9. Grégoire Fleurot, 2013, *Espionnage : Vupen, l'entreprise française qui bosse pour la NSA* [http://www.slate.fr/france/77866/vupen-nsa-espionnage-exploits].
10. Ministry of Justic of Georgia et Al., 2012, *CYBER ESPIONNAGE Against Georgian Government* [http://dea.gov.ge/uploads/CERT%20DOCS/Cyber%20Espionage.pdf] (en anglais).
11. Kevin Poulsen, 2007, *FBI's Secret Spyware Tracks Down Teen Who Made Bomb Threats* [http://www.wired.com/politics/law/news/2007/07/fbi_spyware] (en anglais).
12. Voir les spécifications du logiciel [http://hadopi.fr/download/sites/default/files/page/pdf/Consultation_sur_les%20specifications_fonctionnelles_des_moyens_de_securisation.pdf]

Il est donc fort probable d'en trouver sur le premier Windows que l'on croisera. Jusqu'à présent, utiliser un système d'exploitation minoritaire (tel GNU/Linux) diminue significativement les risques d'infection car ceux-ci sont moins visés, le développement de *malwares* spécifiques étant économiquement moins rentable.

On peut d'ores et déjà évoquer quelques moyens de limiter les risques :

- n'installer (ou n'utiliser) aucun logiciel de provenance inconnue : ne pas faire confiance au premier site web venu [13] ;
- prendre au sérieux les avertissements des systèmes d'exploitation récents qui tentent de prévenir les utilisateurs lorsqu'ils utilisent un logiciel peu sûr, ou lorsqu'ils indiquent qu'une mise à jour de sécurité est nécessaire ;
- enfin, limiter les possibilités d'installation de nouveaux logiciels : en limitant l'utilisation du compte « administrateur » et le nombre de personnes y ayant accès.

3.2 Les matériel espions

page 24

Les adversaires voulant mettre la main sur les secrets contenus par nos ordinateurs peuvent utiliser des logiciels malveillants comme on vient de le voir, mais peuvent tout aussi bien utiliser du matériel espion. Et celui-ci n'a rien à envier au bon vieux James Bond !

Il existe tout une gamme de matériel plus ou moins facilement disponible permettant intrusions ou exfiltration d'information à quasiment tous les niveaux d'un ordinateur. Suite à la révélations de documents confidentiels de la NSA par Edward Snowden, un véritable catalogue du parfait espion informatique a été publié sur le journal allemand Spiegel [14].

Sans en faire un tour exhaustif, on peut découvrir pêle-mêle dans ce catalogue de faux connecteurs USB, permettant de retransmettre sous forme d'ondes radio ce qui transite par ces mêmes connecteurs, des minuscules puces installées dans les câbles reliant écran ou clavier à l'ordinateur et faisant de même, pour qu'un adversaire puisse capter ce qu'on tape ou voit tout en étant à bonne distance. Enfin, pléthore de matériel espion installé dans l'ordinateur, que ce soit au niveau du disque dur, du BIOS, *etc.*

Le tableau n'est pas très encourageant, un véritable audit de son ordinateur demanderait de démonter celui-ci avec très peu de chance de le remonter de telle manière qu'il puisse fonctionner de nouveau. Cela dit, ce matériel n'est pas à la disposition de tous types d'adversaires. De plus, rien n'indique que l'usage d'un tel matériel est devenu monnaie courante, que ce soit pour des raisons de coût, d'installation, ou autres paramètres.

Nous allons quand même nous attarder un peu sur le cas des *keyloggers* qui peuvent entrer à la fois dans la catégorie du matériel espion et des logiciels malveillants.

3.3 Les *keyloggers*, ou enregistreurs de frappe au clavier

Les enregistreurs de frappe au clavier (*keyloggers*), qui peuvent être « matériels » ou « logiciels », ont pour fonction d'enregistrer furtivement tout ce qui est tapé sur un clavier d'ordinateur, afin de pouvoir transmettre ces données à l'agence ou à la personne qui les a installés [15].

13. Ce conseil vaut tout autant pour les personnes utilisant GNU/Linux. En décembre 2009, le site `gnome-look.org` a diffusé un *malware* [`http://lwn.net/Articles/367874/`] présenté comme un économiseur d'écran. Ce dernier était téléchargeable sous forme de paquet Debian au milieu d'autres économiseurs et de fonds d'écran.

14. Spiegel, 2013, *Interactive Graphic : The NSA's Spy Catalog* [`http://www.spiegel.de/international/world/a-941262.html`] (en anglais).

15. security.resist.ca, 2014, *Keystroke Loggers & Backdoors* [`http://security.resist.ca/keylog.shtml`] (en anglais).

Leur capacité à enregistrer touche par touche ce qui est tapé sur un clavier, contournant ainsi tout dispositif de chiffrement, permet d'avoir directement accès aux phrases, mots de passe et autres données sensibles entrées lorsqu'il y a un enregistreur de frappe sur un clavier. page 37

Les *keyloggers* matériels sont des dispositifs reliés au clavier ou à l'ordinateur. Ils peuvent ressembler à des adaptateurs, à des cartes d'extension à l'intérieur de l'ordinateur (PCI ou *mini*-PCI) et même s'intégrer à l'intérieur du clavier [16]. Ils sont donc difficiles à repérer si on ne les recherche pas spécifiquement…

Pour un clavier sans fil, il n'y a même pas besoin de *keylogger* pour récupérer les touches entrées : il suffit de capter les ondes émises par le clavier pour communiquer avec le récepteur, puis de casser le chiffrement utilisé, qui est assez faible dans la plupart des cas [17]. À moindre distance, il est aussi toujours possible d'enregistrer et de décoder les ondes électromagnétiques émises par les claviers avec un fil, y compris ceux qui sont intégrés dans un ordinateur portable… page 14

Les *keyloggers* logiciels sont beaucoup plus répandus, parce qu'ils peuvent être installés à distance (via un réseau, par le biais d'un logiciel malveillant, ou autre), et ne nécessitent généralement pas un accès physique à la machine pour la récupération des données collectées (l'envoi peut par exemple se faire périodiquement par email). La plupart de ces logiciels enregistrent également le nom de l'application en cours, la date et l'heure à laquelle elle a été exécutée ainsi que les frappes de touches associées à cette application.

Aux États-Unis, le FBI utilise depuis de nombreuses années des *keyloggers* logiciels [18].

La seule manière de repérer les *keyloggers* matériels est de se familiariser avec ces dispositifs et de faire régulièrement une vérification visuelle de sa machine, à l'intérieur et à l'extérieur. Même si le catalogue de la NSA publié fin 2013 rend compte de la difficulté de trouver soi-même des dispositifs d'enregistrement de frappe à peine plus gros qu'un ongle. Pour les *keyloggers* logiciels, les pistes sont les mêmes que pour les autres *logiciels malveillants*. page 24

3.4 Des problèmes d'impression ?

On croyait avoir fait le tour des surprises que nous réservent nos ordinateurs… mais même les imprimantes se mettent à avoir leurs petits secrets.

3.4.1 Un peu de stéganographie

Première chose à savoir : de nombreuses imprimantes haut de gamme signent leur travail. Cette signature stéganographique [19] repose sur de très légers détails d'impression, souvent invisibles à l'œil nu, et insérés dans chaque document. Ils permettent d'identifier de manière certaine la marque, le modèle et dans certains cas le numéro de série de la machine qui a servi à imprimer un document. On dit bien « de manière certaine », car c'est pour cela que ces détails sont là : afin de pouvoir retrouver la machine à partir de ses travaux. Toutes les imprimantes ne sont pas pourvues de ce système, baptisé *watermarking*, mais c'est le cas pour nombre de modèles courants [20].

16. Pour se faire une idée, nombre de modèles sont en vente libre [http://www.google.com/products?q=keyloggers] pour une somme allant de 40 à 100 $.
17. ZDNet Australia, 2007, *Microsoft wireless keyboard hacked from 50 metres* [http://www.zdnet.com.au/news/security/soa/Microsoft-wireless-keyboard-hacked-from-50-metres/0,130061744,339284328,00.htm] (en anglais).
18. En 2000, l'usage d'un *keylogger* a permis au FBI [http://www.theregister.co.uk/2000/12/06/mafia_trial_to_test_fbi/] d'obtenir la phrase de passe utilisée par un ponte de la mafia de Philadelphie pour chiffrer ses documents (en anglais).
19. Pour en savoir plus sur la stéganographie, nous conseillons la lecture de cet article Wikipédia, 2014, *Stéganographie* [https://fr.wikipedia.org/wiki/Stéganographie].
20. L'*Electronic Frontier Foundation* tente de maintenir une liste des constructeurs et de ces modèles d'imprimantes indiscrets [http://www.eff.org/issues/printers] (en anglais).

Par ailleurs, d'autres types de traces liées à l'usure de la machine sont aussi laissées sur les documents — et ce avec toutes les imprimantes. Car avec l'âge, les têtes d'impression se décalent, de légères erreurs apparaissent, les pièces s'usent, et tout cela constitue au fur et à mesure une signature propre à l'imprimante. Tout comme la balistique permet d'identifier une arme à feu à partir d'une balle, il est possible d'utiliser ces défauts pour identifier une imprimante à partir d'une page qui en est sortie.

Pour se protéger en partie de cela, il est intéressant de savoir que les détails d'impression ne résistent pas à la photocopie répétée : photocopier la page imprimée, puis photocopier la photocopie obtenue, suffit à faire disparaître de telles signatures. Par contre… on en laissera sûrement d'autres, les photocopieuses présentant des défauts, et parfois des signatures stéganographiques, similaires à ceux des imprimantes. Bref on tourne en rond, et le problème devient surtout de choisir *quelles* traces on veut laisser…

3.4.2 La mémoire, encore…

Certaines imprimantes sont suffisamment « évoluées » pour être plus proches d'un véritable ordinateur que d'un tampon encreur.

page 11

Elles peuvent poser des problèmes à un autre niveau, vu qu'elles sont dotées d'une mémoire vive : celle-ci, tout comme celle du PC, gardera la trace des documents qui ont été traités aussi longtemps que la machine est sous tension… ou jusqu'à ce qu'un autre document les recouvre.

La plupart des imprimantes lasers disposent d'une mémoire vive pouvant contenir une dizaine de pages. Les modèles plus récents ou ceux comportant des scanners intégrés peuvent, quant à eux, contenir plusieurs milliers de pages de texte…

page 12

Pire encore : certains modèles, souvent utilisés pour les gros tirages comme dans les centres de photocopies, disposent parfois de disques durs internes, auxquels l'utilisateur n'a pas accès, et qui gardent eux aussi des traces — et cette fois, même après la mise hors tension.

Chapitre 4

Quelques illusions de sécurité...

Bien. On commence à avoir fait le tour des traces que nous pouvons laisser involontairement, et des informations que des personnes mal intentionnées pourraient nous arracher.

Reste maintenant à pourfendre quelques idées reçues.

4.1 Logiciels propriétaires, *open source*, libres

On a vu qu'un logiciel pouvait faire plein de choses qu'on n'aurait pas du tout envie qu'il fasse. Dès lors, il est indispensable de faire ce que l'on peut pour réduire ce problème autant que possible. De ce point de vue, les logiciels libres sont dignes d'une confiance bien plus grande que les logiciels dits « propriétaires » : nous allons voir pourquoi.

4.1.1 La métaphore du gâteau

Pour comprendre la différence entre logiciels libres et propriétaires, on utilise souvent la métaphore du gâteau. Pour faire un gâteau, il faut une recette : il s'agit d'une liste d'instructions à suivre, des ingrédients à utiliser et d'un procédé de transformation à effectuer. De la même façon, la recette d'un logiciel est appelée « code source ». Elle est écrite dans un langage fait pour être compréhensible par des êtres humains. Cette recette est ensuite transformée en un code compréhensible par le processeur, un peu comme la cuisson d'un gâteau nous donne ensuite la possibilité de le manger.

Les logiciels propriétaires ne sont disponibles que « prêts à consommer », comme un gâteau industriel, sans sa recette. Il est donc très difficile de s'assurer de ses ingrédients : c'est faisable, mais le processus est long et compliqué. Au demeurant, relire une série de plusieurs millions d'additions, de soustractions, de lectures et d'écritures en mémoire pour en reconstituer le but et le fonctionnement est loin d'être la première chose que l'on souhaite faire sur un ordinateur.

Les logiciels libres, au contraire, livrent la recette pour quiconque veut comprendre ou modifier le fonctionnement du programme. Il est donc plus facile de savoir ce qu'on donne à manger à notre processeur, et donc ce qui va s'occuper de nos données.

4.1.2 Les logiciels propriétaires : une confiance aveugle

Un logiciel « propriétaire » est donc un peu comme une « boîte » étanche : on peut constater que le logiciel fait ce qu'on lui demande, possède une belle interface graphique, *etc.* Sauf qu'on ne peut pas vraiment connaître en détail comment il procède ! On ne sait pas s'il se cantonne à faire ce qu'on lui demande, ou s'il fait d'autres choses

en plus. Pour le savoir, il faudrait pouvoir étudier son fonctionnement, ce qui est difficile à faire sans son code source… il ne nous reste donc qu'à lui faire *aveuglément* confiance.

Windows et Mac OS X, les premiers, sont d'immenses boîtes hermétiquement fermées sur lesquelles sont installées d'autres boîtes tout aussi hermétiques (de Microsoft Office aux anti-virus…) qui font peut-être bien d'autres choses que celles qu'on leur demande.

Notamment, balancer des informations que ces logiciels pourraient grapiller sur nous ou permettre d'accéder à l'intérieur de notre ordinateur au moyen de *backdoors*, des « portes dérobées » [1] prévues dans le logiciel pour que ceux qui en ont la clé puissent pirater nos ordinateurs… en fait, vu que l'on ne peut pas savoir comment est écrit le système d'exploitation, on peut tout imaginer en la matière.

Dès lors, laisser reposer la confidentialité et l'intégrité de ses données sur des programmes auxquels on ne peut accorder sa confiance que les yeux fermés, relève de la plus pure illusion de sécurité. Et installer d'autres logiciels prétendant sur leur emballage veiller à cette sécurité à notre place, alors que leur fonctionnement n'est pas plus transparent, ne peut pas résoudre ce problème.

4.1.3 L'avantage d'avoir la recette : les logiciels libres

La confiance plus grande qu'on peut mettre dans un système *libre* comme GNU/Linux est principalement liée au fait de disposer de la « recette » qui permet de le fabriquer. Gardons en tête quand même qu'il n'y a rien de magique : les logiciels libres ne jetent aucun « sort de protection » sur nos ordinateurs.

Toutefois, GNU/Linux offre davantage de possibilités pour rendre un peu plus sûr l'usage des ordinateurs, notamment en permettant de configurer assez finement le système. Ça implique trop souvent des savoirs-faire relativement spécialisés, mais au moins c'est possible.

Par ailleurs, le mode de production des logiciels libres est peu compatible avec l'introduction de portes dérobées : c'est un mode de production collectif, plutôt ouvert et transparent, auquel participent des gens assez variés ; il n'est donc pas facile d'y mettre en toute discrétion des cadeaux à l'attention de personnes mal intentionnées.

Il faut toutefois se méfier des logiciels qualifiés d'*open source*. Ces derniers donnent eux aussi accès à leurs entrailles, mais ont des modes de développement plus fermés, plus opaques. La modification et la redistribution de ces logiciels est au pire interdite, et au mieux autorisée formellement mais rendue en pratique très pénible. Vu que seule l'équipe à l'origine du logiciel va pouvoir participer au développement, on peut considérer que, en pratique, personne ne lira en détail leur code source… et donc que personne ne vérifiera vraiment leur fonctionnement.

C'est le cas par exemple de TrueCrypt, dont le développement s'est arrêté en mai 2014. Logiciel de chiffrement dont le code source est disponible, mais dont le développement est fermé et dont la licence restreint la modification et la redistribution. Pour ce qui nous occupe, le fait qu'un logiciel soit *open source* doit plutôt être considéré comme un argument commercial que comme un gage de confiance.

Sauf que… la distinction entre logiciels libres et *open source* est de plus en plus floue : des employés d'IBM et compagnie écrivent de grosses parties des logiciels libres les plus importants, et on ne va pas toujours regarder de près ce qu'ils écrivent. Par exemple, voici les statistiques des employeurs des gens qui développent le noyau Linux (qui est libre), exprimées en nombre de lignes de code source modifiées, sur une courte période de temps [2] :

1. Au sujet des « portes dérobées » voir l'article Wikipédia, 2014, *Porte dérobée* [https://fr.wikipedia.org/wiki/Porte_dérobée].
2. Linux Weekly News, 2014, *Some 3.14 development statistics* [https://lwn.net/Articles/590354/] (en anglais).

Organisation	Pourcentage
Linux Foundation	11.4%
Intel	10.0%
(None)	6.8%
Red Hat	6.7%
Texas Instruments	4.1%
(Unknown)	3.7%
IBM	3.7%
Realsil Microelectronics	3.4%
SUSE	3.3%
NVIDIA	3.0%
etc.	

Alors… il n'est pas impossible qu'une personne qui écrit un bout de logiciel dans un coin, et à qui la « communauté du libre » fait confiance, ait pu y glisser des bouts de code mal intentionné. Comme ce fut le cas avec la faille connue sous le nom d'*Heartbleed*[3].Si on utilise uniquement des logiciels libres livrés par une distribution GNU/Linux non commerciale, il y a peu de chances que ce cas se présente, mais c'est une possibilité. On fait alors confiance aux personnes travaillant sur la distribution pour étudier le fonctionnement des programmes qui y sont intégrés.

Il est néanmoins important de rappeler que cette confiance ne peut valoir que si on n'installe pas n'importe quoi sur son système. Par exemple, sur Debian, les paquets officiels de la distribution sont « signés », ce qui permet de vérifier leur provenance. Mais si on installe des paquets ou des extensions pour Firefox trouvés sur Internet sans les vérifier, on s'expose à tous les risques mentionnés au sujet des logiciels malveillants. page 24

Pour conclure, et ne pas nous faire plus d'illusions : *libre ou pas, il n'existe pas de logiciel pouvant, à lui seul, assurer l'intimité de nos données*; pour le faire, il n'existe que des pratiques, associées à l'utilisation de *certains logiciels*. Logiciels choisis parce que des éléments nous permettent de leur accorder un certain niveau de confiance.

4.2 Le mot de passe d'un compte ne protège pas ses données

Tous les systèmes d'exploitation récents (Windows, Mac OS X, GNU/Linux) offrent la possibilité d'avoir différents utilisateurs sur un même ordinateur. Il faut bien savoir que les mots de passe qui protègent parfois ces utilisateurs ne garantissent pas du tout la confidentialité des données.

Certes il peut être pratique d'avoir son espace à soi, avec ses propres réglages (marque-pages, fond d'écran…), mais une personne qui souhaiterait avoir accès à toutes les données qu'il y a sur l'ordinateur n'aurait aucun mal à y parvenir : il suffit de rebrancher le disque dur sur un autre ordinateur ou de le démarrer sur un autre système d'exploitation pour avoir accès à toutes les données écrites sur le disque dur. page 15

Aussi, si utiliser des comptes séparés et des mots de passe peut avoir quelques avantages (comme la possibilité de verrouiller l'écran quand on s'éloigne quelques minutes), il est nécessaire de garder en tête que cela ne protège pas réellement les données.

3. Wikipédia, 2014, *Heartbleed* [https://fr.wikipedia.org/wiki/Heartbleed].

4.3 À propos de l'« effacement » des fichiers

page 16 On a déjà évoqué que le contenu d'un fichier devenu inaccessible ou invisible ne s'était pas pour autant volatilisé. On va maintenant détailler pourquoi.

4.3.1 La suppression d'un fichier n'en supprime pas le contenu...

… et ça peut être très facile de le retrouver.

En effet, lorsqu'on « supprime » un fichier — en le plaçant dans la *Corbeille* puis en la vidant — on ne fait que dire au système d'exploitation que le contenu de ce fichier ne nous intéresse plus. Il supprime alors son entrée dans l'index des fichiers existants. Il a ensuite le loisir de réutiliser l'espace que prenaient ces données pour y inscrire autre chose.

Mais il faudra peut-être des semaines, des mois ou des années avant que cet espace soit *effectivement* utilisé pour de nouveaux fichiers, et que les anciennes données disparaissent réellement. En attendant, si on regarde directement ce qui est inscrit sur le disque dur, on retrouve le contenu des fichiers. C'est une manipulation assez simple, automatisée par de nombreux logiciels (qui permettent de « récupérer » ou de « restaurer » des données).

4.3.2 Un début de solution : réécrire plusieurs fois par-dessus les données

Une fois que l'espace d'un disque dur a été réécrit, il devient diffile de retrouver ce qui s'y trouvait auparavant. Mais cela n'est pas pour autant impossible : lorsque l'ordinateur réécrit `1` par-dessus `0`, cela donne plutôt `0,95` et lorsqu'il réécrit `1` par-dessus `1`, cela donne plutôt `1,05`[4]… un peu comme on peut lire sur un bloc-notes ce qui a été écrit sur une page arrachée, par les dépressions créées sur la page vierge située en-dessous.

En revanche ça devient très difficile, voire impossible, de les récupérer quand on réécrit un grand nombre de fois par-dessus, et de différentes manières. La meilleure façon, donc, de rendre inaccessible le contenu de ces fichiers « supprimés », est d'utiliser des logiciels qui s'assureront de le réécrire plusieurs fois, pour terminer par du charabia incompréhensible. C'est ce qu'on appelle « écraser des données » (*wipe* en anglais).

4.3.3 Quelques limites des possibilités de réécriture

Même s'il est possible de réécrire plusieurs fois à un endroit donné d'un disque dur pour rendre inaccessibles les données qu'il contenait, cela ne garantit pas pour autant leur disparition complète du disque…

Les disques « intelligents »

Les disques durs modernes réorganisent leur contenu « intelligemment » : une partie du disque est reservée pour remplacer des endroits qui deviendraient défectueux. Ces opérations de remplacement sont difficilement détectables, et on ne peut jamais être vraiment sûr que l'endroit sur lequel on réécrit trente fois est bien celui où le fichier a été écrit initialement…

Pour les clés USB, on est même sûr que dans la plupart des cas on réécrit à un endroit différent. Comme la mémoire *flash*, utilisée par les clés USB et les disques durs SSD

4. Peter Gutmann, 1996, *Secure Deletion of Data from Magnetic and Solid-State Memory* [http://www.cs.auckland.ac.nz/~pgut001/pubs/secure_del.html] (en anglais).

(*Solid State Disks*), arrête de fonctionner correctement après un certain nombre d'écritures [5], ces derniers contiennent des puces chargées de réorganiser automatiquement le contenu pour répartir les informations au maximum d'endroits différents.

En prenant en compte ces mécanismes, il devient difficile de garantir que les données que l'on souhaite détruire auront bien disparu.

Néanmoins, ouvrir un disque dur pour en examiner les entrailles demande du temps et d'importantes ressources matérielles et humaines… investissement qui ne sera pas forcément à la portée de tout le monde, tout le temps.

Pour les puces de mémoire *flash* d'une clé USB ou d'un disque dur SSD, même si ce n'est pas non plus immédiat, l'opération est beaucoup plus simple : il suffit d'un fer à souder, et d'un appareil permettant de lire directement les puces de mémoire. Ces derniers se trouvent pour environ 1 500 dollars [6].

Les systèmes de fichiers « intelligents »

Un autre problème vient des systèmes de fichiers « intelligents ». page 16

Les systèmes de fichiers développés ces dernières années, comme NTFS ou *ext3*, sont « journalisés », c'est-à-dire qu'ils gardent une trace des modifications successives faites sur les fichiers dans un « journal ». Après une extinction brutale de l'ordinateur, cela permet au système de se contenter de reprendre les dernières opérations à faire, plutôt que de devoir parcourir l'intégralité du disque pour corriger les incohérences. Par contre, cela peut ajouter, encore une fois, des traces sur les fichiers que l'on souhaiterait voir disparaître. page 20

> Le système de fichiers utilisé actuellement le plus souvent sous GNU/Linux, *ext3*, peut fonctionner avec plusieurs modes. Celui le plus courament utilisé ne met dans le journal que les noms des fichiers et d'autres méta-données, pas leur contenu.

D'autres techniques, moins courantes sur un ordinateur personnel, peuvent aussi poser problème : les systèmes de fichiers avec écriture redondante et continuant à écrire même en cas d'erreur, comme les systèmes de fichiers RAID ; les systèmes de fichiers qui effectuent des instantanés (*snapshots*) ; les systèmes de fichiers qui mettent en cache dans des dossiers temporaires, comme les clients NFS (système de fichiers par le réseau) ; les systèmes de fichiers compressés [7].

Enfin, il ne faut pas oublier que le fichier, même parfaitement supprimé, peut avoir laissé des traces ailleurs… page 19

Ce qu'on ne sait pas…

Pour ce qui est des CD-RW ou DVD±RW (ré-inscriptibles), il semble qu'aucune étude sérieuse n'ait été menée à propos de l'efficacité de la réécriture pour rendre des données irrécupérables. Un postulat prudent est donc de détruire méthodiquement les supports de ce type qui auraient pu contenir des données à faire disparaître.

5. Les modèles bas de gamme ne fonctionneront plus correctement après avoir été écrits cent mille fois, et cinq millions pour les meilleurs. D'après Wikipédia, 2014, *Solid-state drive* [https://fr.wikipedia.org/wiki/Solid_State_Drive].

6. Le Salvation Data Flash Doctor [http://www.sd-flash.com/] ou encore le PC-3000 Flash SSD Edition [http://www.pc-3000flash.com/] sont tous les deux vendus comme des *outils professionnels de recouvrement de données sur des périphériques flash endommagés* (liens en anglais).

7. Page de manuel de `shred(1)` [http://manpages.debian.net/cgi-bin/man.cgi?query=shred&locale=fr].

4.3.4 Plein d'autres fois où l'on « efface »

Il faut noter qu'on ne supprime pas seulement des fichiers en les mettant à la corbeille. Par exemple, quand on utilise l'option « Effacer mes traces » du navigateur Firefox, ce dernier ne fait pas mieux que de *supprimer* les fichiers. Certes les données sont devenues inaccessibles pour Firefox, mais elles sont toujours accessibles en regardant directement le disque dur.

Enfin, il est utile d'insister ici sur le fait que le *reformatage* d'un disque dur n'efface pas pour autant le contenu qui s'y trouvait. De même que la suppression des fichiers, cela ne fait que rendre disponible l'espace où se trouvait le contenu précédemment, les données restant physiquement présentes sur le disque. Tout comme détruire le catalogue d'une bibliothèque ne fait pas pour autant disparaître les livres présents dans les rayonnages…

On peut donc toujours retrouver des fichiers après un reformatage, aussi facilement que s'ils avaient été simplement « supprimés »…

4.3.5 Et pour ne laisser aucune trace ?

Pour régler radicalement le problème, il n'y a pas de méthode simple. La solution la moins difficile pour l'instant est d'utiliser l'ordinateur après l'avoir démarré avec un système *live* configuré pour n'utiliser que la mémoire vive, comme *Tails*. Alors, il est possible de ne rien écrire sur le disque dur ni sur le *swap*, et de ne garder les informations (tant que l'ordinateur est allumé) que dans la mémoire vive.

4.4 Les logiciels portables : une fausse solution

Ce qu'on appelle « logiciels portables », ce sont des logiciels qui ne sont pas installés sur un système d'exploitation donné, mais que l'on peut démarrer depuis une clé USB ou un disque dur externe — et donc, transporter avec soi afin d'en disposer sur n'importe quel ordinateur.

Il est devenu très facile de télécharger sur Internet de telles applications. Des « packs portables » ont ainsi été mis en ligne, comme Firefox avec Tor, ou Thunderbird avec Enigmail.

Toutefois, contrairement aux systèmes *live*, ils se servent du système d'exploitation installé sur l'ordinateur où on les utilise (la plupart du temps, ils sont prévus pour Windows).

L'idée qui est à leur origine est de permettre d'avoir toujours les logiciels dont on a besoin, sous la main, personnalisés à notre usage. Mais « transporter son bureau partout avec soi », par exemple, n'est pas forcément la meilleure manière de préserver la confidentialité de ses données.

Disons-le tout de suite : ces logiciels ne protègent pas plus les personnes qui s'en servent que des logiciels « non portables ». Pire, le discours faisant leur promotion participe à créer une illusion de sécurité avec d'énormes bêtises comme « *l'utilisation des logiciels se fait de façon sécurisée et sans laisser d'informations personnelles sur les machines sur lesquelles vous utilisez votre Framakey.* » [8]. C'est malheureusement faux.

8. Cet extrait provient de la page d'accueil du site web de FramaKey [http://www.framakey.org/], une compilation de logiciels portables réalisée par Framasoft [http://www.framasoft.net/], un site français de promotion du logiciel libre. Sur la présentation de la FramaKey [http://framakey.org/Main/Details], on peut lire maintenant « *le navigateur web et le client mail protègeront votre intimité et l'ordinateur hôte en laissant un minimum de traces* »… sans plus de précisions sur la nature de ces traces.

4.4.1 Principaux problèmes

Ces solutions « clé en main » posent donc quelques problèmes plutôt fâcheux…

Il restera des traces sur le disque dur

Si le logiciel a été rendu « portable » correctement, il ne devrait pas laisser délibérément de traces sur le disque dur de l'ordinateur sur lequel on l'utilise. Mais en fait, le logiciel n'a jamais un contrôle absolu. Il dépend en effet largement du système d'exploitation sur lequel il est employé, qui peut avoir besoin d'écrire de la « mémoire virtuelle » sur le disque dur, ou d'enregistrer diverses traces de ce qu'il fait dans ses journaux et autres « documents récents ». Tout cela restera ensuite sur le disque dur.

page 19
page 20
page 21

Il n'y a aucune raison d'avoir confiance en un système inconnu

On a vu auparavant que beaucoup de systèmes ne faisaient absolument pas ce que l'on croit. Or, puisque le logiciel portable va utiliser le système installé sur l'ordinateur sur lequel on le lance, on souffrira de tous les mouchards et autres logiciels malveillants qui pourraient s'y trouver…

page 23

On ne sait pas qui les a compilés, ni comment

Les modifications apportées aux logiciels pour les rendre portables sont rarement vérifiées, alors même qu'elles ne sont généralement pas faites par les auteurs du logiciel lui-même. Dès lors, on peut soupçonner ces logiciels, encore plus que leurs versions non-portables, de contenir des failles de sécurité, qu'elles aient été introduites par erreur ou volontairement.

On traitera plus loin de la question de l'hygiène minimale à avoir dans le choix des logiciels qu'on installe ou télécharge.

Chapitre 5

Une piste pour se protéger : la cryptographie

La *cryptographie* est la branche des mathématiques qui s'occupe spécifiquement de protéger des messages. Jusqu'en 1999, l'usage de techniques cryptographiques était interdit au grand public. C'est devenu légal entre autres pour permettre aux services marchands sur Internet de se faire payer sans que les clients se fassent piquer leur numéro de carte bleue.

La *cryptanalyse* est le domaine consistant à « casser » les techniques cryptographiques, par exemple pour permettre de retrouver un message qui avait été protégé [1].

Lorsque l'on veut protéger des messages, on distingue trois aspects :

- confidentialité : empêcher les regards indiscrets ;
- authenticité : s'assurer de la source du message ;
- intégrité : s'assurer que le message n'a pas subi de modification.

On peut désirer ces trois choses en même temps, mais on peut aussi vouloir seulement l'une ou l'autre. L'émetteur d'un message *confidentiel* peut souhaiter nier en être l'auteur (et donc qu'on ne puisse pas l'*authentifier*). On peut aussi imaginer vouloir certifier la provenance (*authentifier*) et l'*intégrité* d'un communiqué officiel qui sera diffusé publiquement (donc loin d'être *confidentiel*).

Dans tout ce qui suit, on va parler de *messages*, mais les techniques cryptographiques s'appliquent de fait à n'importe quels nombres, donc à n'importe quelles données, une fois numérisées.

À noter, la cryptographie ne cherche pas à cacher les messages, mais à les protéger. Pour cacher des messages, il est nécessaire d'avoir recours à des techniques stéganograpiques (comme celles utilisées par les imprimantes évoquées auparavant), dont nous ne parlerons pas ici. page 27

5.1 Protéger des données des regards indiscrets

Comme l'ont bien compris les gamins utilisant des codes pour s'échanger des messages ou les militaires communiquant leurs ordres, la piste la plus sérieuse pour que des données ne puissent être comprises que par les personnes « dans le secret », c'est celle du *chiffrement*.

Le chiffrement d'un fichier ou d'un support de stockage permet de le rendre illisible pour toute personne qui n'a pas le code d'accès (souvent une *phrase de passe*). Il sera

1. Pour un bon aperçu des différentes méthodes, qu'on appelle des « attaques », couramment utilisées en cryptanalyse, on peut se référer à la page Wikipédia, 2014, *Cryptanalyse* [https://fr.wikipedia.org/wiki/Cryptanalyse].

certes toujours possible d'accéder au contenu, mais les données ressembleront à une série de nombres aléatoires, et seront donc illisibles.

Souvent on dit *crypter* et *décrypter* à la place de *chiffrer* et *déchiffrer*, ce qui peut porter à confusion ; les termes sont cependant synonymes.

5.1.1 Comment ça marche ?

Grosso modo, il y a seulement trois grandes idées pour comprendre comment on peut chiffrer des messages [2].

La première idée : la *confusion*. Il faut obscurcir la relation entre le message originel et le message chiffré. Un exemple très simple est le « chiffre de César » :

texte en clair	:	ASSAUT	DANS	UNE	HEURE
		↓↓↓↓↓↓	↓↓↓↓	↓↓↓	↓↓↓↓↓
texte chiffré	:	DVVDXW	GDQV	XQH	KHXUH

A + 3 lettres = D

Sauf qu'avec le chiffre de César, il est facile d'analyser la fréquence des lettres et de retrouver les mots.

Alors la deuxième grande idée, c'est la *diffusion*. Cela permet d'éclater le message pour le rendre plus difficile à reconnaître. Un exemple de cette technique, c'est la transposition par colonne :

$$\begin{pmatrix}A\\D\\E\end{pmatrix}\begin{pmatrix}S\\A\\H\end{pmatrix}\begin{pmatrix}S\\N\\E\end{pmatrix}\begin{pmatrix}A\\S\\U\end{pmatrix}\begin{pmatrix}U\\U\\R\end{pmatrix}\begin{pmatrix}T\\N\\E\end{pmatrix} \xrightarrow[\text{diffusion en 3 points}]{} \begin{matrix}\text{ADE SAH SNE}\\ \text{ASU UUR TNE}\end{matrix}$$

Ce que l'on appelle un *algorithme de chiffrement*, ce sont les différentes techniques utilisées pour transformer le texte original. Quant à la *clé* de chiffrement, c'est, par exemple dans le cas du chiffre de César, le nombre de caractères de décalage (3 en l'occurence), ou dans la technique de diffusion, le nombre de lignes des colonnes. La valeur de cette clé est variable, on aurait tout aussi bien décider de faire des colonnes de 2 lignes, ou un décalage de 6 caractères.

Ce qui nous amène à la troisième grande idée : *le secret réside seulement dans la clé*. Après quelques millénaires, on s'est aperçu que c'était une mauvaise idée de partir du principe que personne n'arriverait à comprendre l'algorithme de chiffrement. Tôt ou tard, une personne finira bien par le découvrir… par la force si nécessaire.

De nos jours, l'algorithme peut donc être détaillé sur Wikipédia en long, en large et en travers, permettant à n'importe qui de vérifier qu'il n'a pas de point faible particulier, c'est-à-dire que la seule solution pour déchiffrer un texte sera de disposer de la *clé* qui a été employée avec celui-ci.

5.1.2 Vous voulez un dessin ?

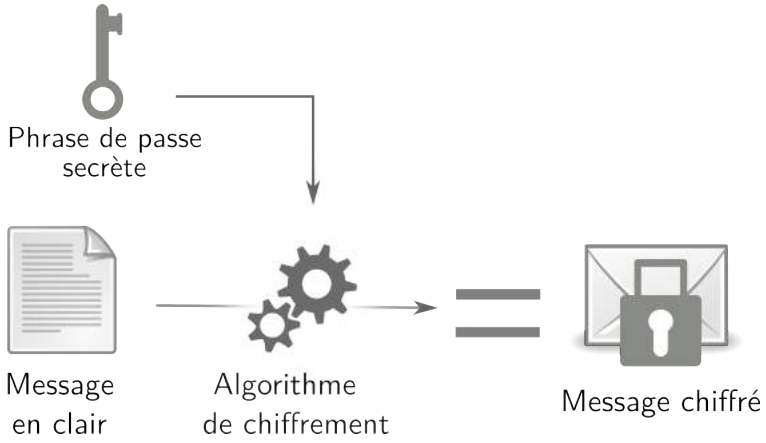
Concrètement, pour assurer la *confidentialité* de nos données, on utilise deux opérations : le chiffrement, puis le déchiffrement.

Première étape : le chiffrement

Pour un exemple d'usage pratique, prenons le message suivant [3] :

2. Le passage qui suit est une adaptation très partielle de la bande dessinée de Jeff Moser sur l'algorithme AES [http://www.moserware.com/2009/09/stick-figure-guide-to-advanced.html] (en anglais).

3. Ce message est d'une très haute importance stratégique pour des personnes qu'on inviterait chez soi. Il est donc crucial de le chiffrer.




```
Les spaghetti sont dans le placard.
```

Après avoir chiffré ce message en utilisant le logiciel GnuPG avec l'algorithme AES256, et comme phrase de passe « *ceci est un secret* », on obtient :

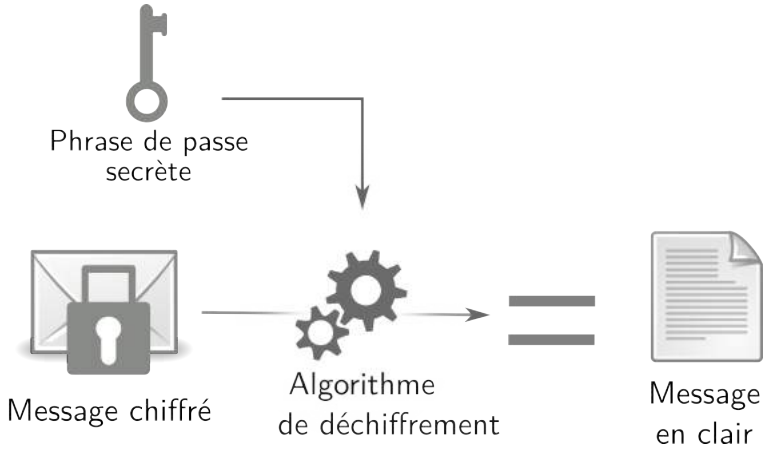
```
-----BEGIN PGP MESSAGE-----

jA0ECQMCRM0lmTSIONRg0lkBWGQI76cQOocEvdBhX6BM2AU6aYSPYymSqj8ihFXu
wV1GVraWuwEt4XnLc3F+OxT3EaXINMHdH9oydA92WDkaqPEnjsWQs/oSCeZ3WXoB
9mf9y6jzqozEHw==
=T6eN
-----END PGP MESSAGE-----
```

Voici donc l'aspect que prend un texte après chiffrement : son contenu est devenu parfaitement imbuvable. Les données « en clair », lisibles par tout le monde, ont été transformées en un autre format, incompréhensible pour qui ne possède pas la clé.

Deuxième étape : le déchiffrement

Pour le déchiffrement, il nous suffira d'utiliser de nouveau GnuPG, avec notre texte chiffré, cette fois. GnuPG nous demandera la phrase de passe ayant servi à chiffrer notre texte, et si cette dernière est correcte, on obtiendra enfin l'information qui nous manquait pour préparer le déjeuner.



5.1.3 Pour un disque dur…

Si on souhaite mettre sur un support de stockage (disque dur, clé USB, *etc.*) uniquement des données chiffrées, il va falloir que le système d'exploitation se charge de réaliser « à la volée » les opérations de chiffrement et de déchiffrement.

Ainsi, chaque fois que des données devront êtres lues du disque dur, elles seront déchiffrées au passage afin que les logiciels qui en ont besoin puissent y accéder. À l'inverse, chaque fois qu'un logiciel demandera à écrire des données, elles seront chiffrées avant d'atterrir sur le disque dur.

page 11

Pour que ces opérations fonctionnent, il est nécessaire que la clé de chiffrement se trouve en mémoire vive aussi longtemps que le support aura besoin d'être utilisé.

Par ailleurs, la clé de chiffrement ne peut pas être changée. Une fois que cette dernière a servi à chiffrer des données inscrites sur le disque, elle devient indispensable pour pouvoir les relire. Pour pouvoir changer la clé, il faudrait donc relire puis réécrire l'intégralité des données du disque…

Pour éviter cette opération pénible, la plupart des systèmes utilisés pour chiffrer les supports de stockage utilisent donc une astuce : la clé de chiffrement est en fait un grand nombre, totalement aléatoire, qui sera lui-même chiffré à l'aide d'une *phrase de passe*[4]. Cette version chiffrée de la clé de chiffrement est généralement inscrite sur le support de stockage au début du disque, « *en tête* » des données chiffrées.

Avec ce système, changer le code d'accès devient simple, vu qu'il suffira de remplacer uniquement cet *en-tête* par un nouveau.

5.1.4 Résumé et limites

La cryptographie permet donc de bien protéger ses données[5], en chiffrant tout ou partie de son disque dur comme de tout autre support de stockage (clé USB, CD, *etc.*), ou de ses communications. De plus, les ordinateurs modernes sont suffisamment puissants pour que nous puissions faire du chiffrement une routine, plutôt que de le réserver à des circonstances spéciales ou à des informations particulièrement sensibles (sinon, cela identifie tout de suite ces dernières comme importantes, alors qu'il vaut mieux les dissoudre dans la masse).

tome 2 ch. 6

On peut ainsi mettre en place une phrase de passe pour chiffrer tout un disque dur, et/ou donner à certaines personnes une partie chiffrée avec leur propre phrase de passe. Il est également possible de chiffrer individuellement tel ou tel fichier, ou un email, ou une pièce jointe, avec une phrase de passe encore différente.

Cependant, bien qu'il soit un outil puissant et essentiel pour la sécurité des informations, **le chiffrement a ses limites** — en particulier lorsqu'il n'est pas utilisé correctement.

Comme expliqué auparavant, lorsqu'on accède à des données chiffrées, il est nécessaire de garder deux choses en tête. Premièrement, une fois les données déchiffrés, ces dernières se trouvent *au minimum* dans la mémoire vive. Deuxièment, tant que des données doivent être chiffrées ou déchiffrées, la mémoire vive contient également la *clé de chiffrement*.

Toute personne qui dispose de la clé de chiffrement pourra lire *tout ce qui a été chiffré avec*, et aussi s'en servir pour chiffrer elle-même des données.

Il faut donc faire attention aux éléments suivants :

4. Le système LUKS, utilisé sous GNU/Linux, permet même d'utiliser plusieurs versions chiffrées de la clé de chiffrement. Chacune de ces versions pourra être chiffrée avec une *phrase de passe* différente, ce qui permet à plusieurs personnes d'accéder aux mêmes données sans pour autant avoir à retenir le même secret.

5. Un article en anglais sur un procès où la police avoue son impuissance face au chiffrement : Philip Willan, 2003, *PGP Encryption Proves Powerful* [http://www.pcworld.com/article/110841/article.html]

- Le système d'exploitation et les logiciels ont accès aux données et à la clé de chiffrement autant que nous, alors cela dépend de la confiance qu'on met en eux — encore une fois, il s'agit de ne pas installer n'importe quoi n'importe comment. page 24
- Quiconque obtient un accès physique à l'ordinateur allumé a, de fait, accès au contenu de la mémoire vive. Lorsqu'un disque chiffré est activé, celle-ci contient, en clair, les données sur lesquelles on a travaillé depuis l'allumage de l'ordinateur (même si elles sont chiffrées sur le disque). Mais elle contient surtout, comme dit plus haut, la clé de chiffrement, qui peut donc être recopiée. Donc il vaut mieux s'habituer, quand on ne s'en sert pas, à éteindre les ordinateurs, et à désactiver (démonter, éjecter) les disques chiffrés. page 19
- Dans certains cas, il peut être nécessaire de prévoir des solutions matérielles pour pouvoir couper le courant facilement et rapidement[6] ; ainsi les disques chiffrés redeviennent inaccessibles sans la phrase de passe — à moins d'effectuer une *cold boot attack*. page 19
- Il reste également possible qu'un enregistreur de frappe ait été installé sur l'ordinateur, et que celui-ci enregistre la phrase de passe. page 26

Par ailleurs, une certaine limite « légale » vient s'ajouter aux possibles attaques. En France, toute personne qui chiffre ses données est en effet censée donner le code d'accès aux autorités lorsqu'elles le demandent, comme l'explique l'*article 434-15-2 du Code Pénal*[7] :

> Est puni de trois ans d'emprisonnement et de 45 000 euros d'amende le fait, pour quiconque ayant connaissance de la convention secrète de déchiffrement d'un moyen de cryptologie susceptible d'avoir été utilisé pour préparer, faciliter ou commettre un crime ou un délit, de refuser de remettre ladite convention aux autorités judiciaires ou de la mettre en œuvre, sur les réquisitions de ces autorités délivrées en application des titres II et III du livre Ier du code de procédure pénale.
>
> Si le refus est opposé alors que la remise ou la mise en œuvre de la convention aurait permis d'éviter la commission d'un crime ou d'un délit ou d'en limiter les effets, la peine est portée à cinq ans d'emprisonnement et à 75 000 euros d'amende.

À noter là-dedans : *susceptible* et *sur les réquisitions*. C'est-à-dire que la loi est assez floue pour permettre d'exiger de toute personne détentrice de données chiffrées qu'elle crache le morceau. On peut éventuellement se voir demander la phrase de passe d'un support qui ne serait pas le nôtre… et que nous n'aurions donc pas. On notera que personne, à notre connaissance, n'a pour l'instant jamais été condamné pour ça.

Enfin, il peut être judicieux de rappeler que les mathématiques utilisées dans les algorithmes cryptographiques ont parfois des défauts. Et beaucoup plus souvent encore, les logiciels qui les appliquent comportent des faiblesses. Certains de ces problèmes peuvent, du jour au lendemain, transformer ce qu'on pensait être la meilleure des protections en une simple affaire de « double clic »…

5.2 S'assurer de l'intégrité de données

Nous avons vu quelques pistes pour assurer la *confidentialité* de nos données. Toutefois, il peut être aussi important de pouvoir s'assurer de leur *intégrité*, c'est-à-dire de vérifier qu'elles n'aient pas subi de modification (par accident ou par malveillance).

6. Pour cette raison, il est de bon ton de ne pas laisser la batterie branchée dans un ordinateur portable quand elle n'est pas utilisée. Il suffit alors d'enlever le câble secteur pour l'éteindre.

7. Le terme légal est « cryptologie ». Une recherche sur ce mot sur Légifrance [http://www.legifrance.gouv.fr] donnera une liste exhaustive des textes de loi concernant ce domaine.

On peut également vouloir s'assurer de la provenance de nos données, en assurer l'*authenticité*.

Concrètement, après la lecture de ces pages, on peut comprendre à quel point il est complexe de s'assurer que les logiciels que l'on souhaite installer sur nos ordinateurs n'aient pas été modifiés en route afin d'y introduire des logiciels malveillants.

page 24

5.2.1 La puissance du hachoir

L'essentiel des techniques pour assurer l'intégrité ou l'authenticité reposent sur des outils mathématiques que la cryptographie a baptisés « fonctions de hachage ».

Ces dernières fonctionnent comme des *hachoirs*, capables de réduire n'importe quoi en tout petits morceaux. Et si notre hachoir fonctionne bien pour être utilisé en cryptographie, on sait que :

- avec les petits morceaux, impossible de reconstituer l'objet original sans essayer tous les objets de la terre ;
- le même objet, une fois passé au hachoir, donnera toujours les mêmes petits morceaux ;
- deux objets différents doivent donner des petits morceaux différents.

Lorsque ces propriétés sont réunies, il nous suffit alors de comparer les petits morceaux issus de deux objets pour savoir s'il étaient identiques.

Les petits morceaux qui sortent de notre hachoir s'appellent plus couramment une *somme de contrôle* ou une *empreinte*. Elle est généralement écrite sous une forme qui ressemble à :

```
f9f5a68a721e3d10baca4d9751bb27f0ac35c7ba
```

Vu que notre hachoir fonctionne avec des données de n'importe quelle taille et de n'importe quelle forme, comparer des empreintes peut nous permettre de comparer plus facilement des images, des CD, des logiciels, *etc.*

Notre hachoir n'est pas magique pour autant. On imagine tout de même bien qu'en réduisant n'importe quoi en petits cubes de taille identique, on peut se retrouver avec les mêmes petits cubes issus de deux objets différents. Cela s'appelle une *collision*. Ce carambolage mathématique n'est heureusement dangereux que lorsqu'il est possible de le provoquer… ce qui est déjà arrivé pour plusieurs fonctions de hachage après quelques années de recherche.

5.2.2 Vérifier l'intégrité d'un logiciel

Prenons un exemple : Alice a écrit un programme et le distribue sur des CD, que l'on peut trouver dans des clubs d'utilisateurs de GNU/Linux. Betty a envie d'utiliser le programme d'Alice, mais se dit qu'il aurait été très facile pour une administration mal intentionnée de remplacer un des CD d'Alice par un logiciel malveillant.

Elle ne peut pas aller chercher un CD directement chez Alice, qui habite dans une autre ville. Par contre, elle a rencontré Alice il y a quelque temps, et connaît sa voix. Elle lui téléphone donc, et Alice lui donne la *somme de contrôle* du contenu du CD :

$$\text{CD d'Alice} \xrightarrow[\text{SHA256}]{} \begin{matrix}\texttt{94d93910609f65475a189d178ca6a45f}\\ \texttt{22b50c95416affb1d8feb125dc3069d0}\end{matrix}$$

Betty peut ensuite la comparer avec celle qu'elle génère à partir du CD qu'elle s'est procuré :

CD de Betty $\xrightarrow[\text{SHA256}]{}$ `94d93910609f65475a189d178ca6a45f`
`22b50c95416affb1d8feb125dc3069d0`

Comme les nombres sont les mêmes, Betty est contente, elle est sûre de bien utiliser le même CD que celui fourni par Alice.

Calculer ces sommes de contrôle ne leur prend pas beaucoup plus de temps que la lecture complète du CD… soit quelques minutes tout au plus.

Maintenant, mettons-nous dans la peau de Carole, qui a été payée pour prendre le contrôle de l'ordinateur de Betty à son insu. Pour cela, elle veut créer un CD qui ressemble à celui d'Alice, mais qui contient un logiciel malveillant.

Malheureusement pour elle, la fonction de hachage ne va que dans un sens. Elle doit donc commencer par se procurer le CD original d'Alice. page précédente

Ensuite, elle modifie ce CD pour y introduire le logiciel malveillant. Cette première version ressemble de très près à l'original. Cela pourrait duper plus d'une personne qui ne ferait pas attention, mais elle sait que Betty verifiera la somme de contrôle du CD qui lui permettra d'installer la nouvelle version.

Comme Alice utilise la fonction de hachage SHA256, qui n'a pas de défaut connu, il ne reste à Carole qu'à essayer un très grand nombre de variation des données de son CD, cela dans l'espoir d'obtenir une *collision*, c'est-à-dire la même somme de contrôle que celle d'Alice.

Malheureusement pour elle, et heureusement pour Betty, même avec de nombreux ordinateurs puissants, les chances de réussite de Carole dans un temps raisonnable (mettons, quelques années) sont extrêmement faibles.

Il suffit donc de se procurer une *empreinte*, ou *somme de contrôle*, par des intermédiaires de confiance pour vérifier l'*intégrité* de données. Tout l'enjeu est ensuite de se procurer ces empreintes par un moyen de confiance : de pouvoir vérifier leur *authenticité*…

5.2.3 Vérifier un mot de passe

Un autre exemple d'utilisation des fonctions de hachage concerne la vérification de l'*authenticité* d'une demande d'accès.

Si l'accès à un ordinateur est protégé par un mot de passe, comme l'ouverture d'une session sous GNU/Linux[8], il faut que l'ordinateur puisse vérifier que le mot de passe est le bon. Mais les mots de passe ne sont pas enregistrés sur l'ordinateur, car il serait trop facile de les lire.

Mais alors comment l'ordinateur s'assure-t-il que le mot de passe tapé au clavier est exact ?

Lorsque l'on choisit un mot de passe pour son ordinateur, le système enregistre en fait, grâce à une fonction de hachage, une empreinte du mot de passe. Pour vérifier l'accès, il « hache » de la même manière le mot de passe que l'on a saisi. Et si les empreintes sont les mêmes, il considère que le mot de passe était le bon.

Il est donc possible de vérifier que le mot de passe correspond, sans garder le mot de passe lui-même !

5.3 Symétrique, asymétrique ?

Les techniques de chiffrement mentionnées jusqu'ici reposent sur une seule clé secrète, qui permet à la fois d'effectuer le chiffrement et le déchiffrement. On parle dans ce cas de cryptographie *symétrique*.

8. Rappelons-nous que ces mots de passe ne servent pas à protéger les données [page 31] !

Ceci en opposition avec la cryptographie *asymétrique* qui n'utilise pas la même clé pour chiffrer et déchiffrer. Autrement appelé « chiffrement à clé publique », ce dernier est surtout utilisé pour la communication « en ligne », on en parlera donc en détail dans le prochain tome.

tome 2 ch. 6

Une des propriétés les plus intéressantes de la cryptographie asymétrique que l'on peut évoquer brièvement est la possibilité de réaliser des *signatures numériques*. Comme son équivalent papier, une signature numérique permet d'apposer une marque de reconnaissance sur des données.

Ces signatures numériques utilisant la cryptographie asymétrique constituent la façon la plus simple de vérifier la provenance d'un logiciel. On sera donc amené à s'en servir plus loin…

Deuxième partie

Choisir des réponses adaptées

La panique s'est désormais emparée de nous. Tout ce qu'on fait sur un ordinateur nous trahit, jour après jour. Qui plus est lorsqu'on croit, à tort, « être en sécurité ».

Mais avant de retourner au pigeon voyageur et à la cache secrète derrière la bibliothèque, qu'on ouvre en tirant sur un faux livre (solutions rustiques à ne pas oublier totalement, ceci dit…), il y a un peu de marge. Pas tant que ça, mais tout de même.

C'est cette marge que ce texte s'appliquera dorénavant à cartographier.

Dans cette partie, c'est en expliquant quelques idées, tout aussi importantes qu'elles sont générales, que nous brosserons le tableau d'une méthodologie sommaire permettant à quiconque de répondre à la question suivante : *comment décider d'un ensemble de pratiques et d'outils adéquats à notre situation ?* Nous décrirons ensuite quelques situations-types, que nous nommons des *cas d'usage*, afin d'illustrer notre propos.

Chapitre 6

Évaluation des risques

Quand on se demande quelles mesures mettre en place pour protéger des données ou des communications numériques, on se rend assez vite compte qu'en la matière, on avance un peu à l'aveuglette.

D'abord parce que la plupart des solutions qu'on pourrait mettre en place ont aussi leurs inconvénients : parfois elles sont très pénibles à déployer, à entretenir ou à utiliser ; parfois on a le choix entre diverses techniques, dont aucune ne répond complètement au « cahier des charges » que l'on s'est fixé ; parfois elles sont bien trop nouvelles pour avoir l'assurance qu'elles fonctionnent réellement ; *etc.*

6.1 Que veut-on protéger ?

Dans le cadre de ce texte, ce qu'on veut protéger rentre en général dans la vaste catégorie de l'*information* : par exemple, le contenu de messages électroniques, des fichiers de données (photo, tracts, carnet d'adresses) ou l'existence même d'une correspondance entre telle et telle personne.

Le mot « protéger » recouvre différents besoins :

- **confidentialité** : cacher des informations aux yeux indésirables ;
- **intégrité** : conserver des informations en bon état, et éviter qu'elles ne soient modifiées sans qu'on s'en rende compte ;
- **accessibilité** : faire en sorte que des informations restent accessibles aux personnes qui en ont besoin.

Il s'agit donc de définir, pour chaque ensemble d'informations à protéger, les besoins de confidentialité, d'intégrité et d'accessibilité. Sachant que ces besoins entrent généralement en conflit, on réalise dès maintenant qu'il faudra, par la suite, poser des priorités et trouver des compromis entre eux : en matière de sécurité informatique, on a rarement le beurre et l'argent du beurre.

6.2 Contre qui veut-on se protéger ?

Rapidement, se pose la question des capacités des personnes qui en auraient après ce que l'on veut protéger. Et là, ça se corse, parce qu'il n'est par exemple pas facile de savoir ce que les personnes les plus qualifiées peuvent réellement faire, et de quels moyens et de quels budgets elles bénéficient. En suivant l'actualité, et par divers autres biais, on peut se rendre compte que cela varie beaucoup selon à qui on a affaire. Entre le gendarme du coin et la *National Security Agency* américaine, il y a tout un fossé sur les possibilités d'actions, de moyens et de techniques employées.

Par exemple, le chiffrement est un des moyens les plus adaptés pour éviter qu'une personne qui allumerait, déroberait ou saisirait judiciairement un ordinateur accède à

page 37

toutes les données qui y résident. Mais les lois en vigueur en France ont prévu le coup : dans le cadre d'une enquête, toute personne doit donner la clé de chiffrement afin de permettre aux enquêteurs d'avoir accès aux données, sans quoi elle risque des peines assez lourdes. Cette loi permet à des enquêteurs ayant peu de moyens techniques d'agir contre ce type de protection, même si en réalité, nous ne connaissons aucun cas où cette loi a été appliquée. En parallèle, des organismes disposent de plus de moyens, tels la NSA ou la DGSE, et rien n'est sûr concernant leurs possibilités. Quelle avance ont-ils dans le domaine du cassage de cryptographie ? Sont-ils au courant de failles dans certaines méthodes, qu'ils n'auraient pas dévoilées, et qui leur permettraient de lire les données ? Sur ces sujets, il n'y a évidemment aucun moyen d'être sûr de ce que ces entités peuvent faire, mais en même temps leur champ d'intervention est limité, et il y a peu de cas dans lesquels on risque d'être confronté à elles.

Un facteur important est aussi à prendre en compte : le coût. En effet, plus les moyens mis en place sont importants, plus les technologies utilisées sont complexes, et plus leur coût est élevé ; ça signifie qu'ils ne seront utilisés que dans des cas précis et tout aussi importants aux yeux des personnes concernées. Par exemple, il y a peu de chances de voir un ordinateur soumis à d'intenses tests dans de coûteuses expertises pour une affaire de vol à l'étalage.

Dès lors, avant même de chercher une solution, la question est de savoir qui pourrait tenter d'accéder à nos informations sensibles, afin de discerner s'il est nécessaire de chercher des solutions compliquées ou pas. Sécuriser complètement un ordinateur est de toutes façons de l'ordre de l'impossible, et dans cette histoire, il s'agit plutôt de mettre des bâtons dans les roues de celles et ceux qui pourraient en avoir après ce que l'on veut protéger. Plus l'on pense grands les moyens de ces personnes, plus les bâtons doivent être nombreux et solides.

Évaluer les risques, c'est donc avant tout se poser la question de quelles sont les données que l'on veut protéger, et de qui peut être intéressé par ces données. À partir de là, on peut avoir une vision de quels moyens ils disposent (ou en tout cas, dans la mesure du possible, essayer de se renseigner) et en conséquence, définir une *politique de sécurité* adaptée.

Chapitre 7

Définir une politique de sécurité

Une chaîne n'a que la solidité de son maillon le plus faible. Rien ne sert d'installer trois énormes verrous sur une porte blindée placée à côté d'une frêle fenêtre délabrée. De même, chiffrer une clé USB ne rime pas à grand-chose si les données qui y sont stockées sont utilisées sur un ordinateur qui en conservera diverses traces *en clair* sur son disque dur.

page 37
page 19

Ces exemples nous apprennent quelque chose : de telles « solutions » ciblées ne sont d'aucune utilité tant qu'elles ne font pas partie d'un ensemble de pratiques articulées de façon cohérente. Qui plus est, les informations qu'on veut protéger sont le plus souvent en relation avec des pratiques hors du champ des outils numériques. C'est donc de façon globale qu'il faut évaluer les risques et penser les réponses adéquates.

page 49

De façon globale, mais *située* : à une situation donnée correspond un ensemble singulier d'enjeux, de risques, de savoirs-faire… et donc de possibilités d'action. Il n'existe pas de solution miracle convenant à tout le monde, et qui réglerait tous les problèmes d'un coup de baguette magique. La seule voie praticable, c'est d'en apprendre suffisamment pour être capables d'imaginer et de mettre en place une politique de sécurité adéquate à sa propre situation.

7.1 Une affaire de compromis

On peut toujours *mieux* protéger ses données et ses communications numériques. Il n'y a de limite ni aux possibilités d'attaque et de surveillance, ni aux dispositifs qu'on peut utiliser pour s'en protéger. Cependant, à chaque protection supplémentaire qu'on veut mettre en place correspond un effort en termes d'apprentissage, de temps ; non seulement un effort initial pour s'y mettre, pour installer la protection, mais aussi, bien souvent, une complexité d'utilisation supplémentaire, du temps passé à taper des phrases de passe, à effectuer des procédures pénibles et répétitives, à porter son attention sur la technique plutôt que sur l'usage qu'on voudrait avoir de l'ordinateur.

Dans chaque situation, il s'agit donc de trouver un **compromis** convenable entre la facilité d'utilisation et le niveau de protection souhaité.

Parfois, ce compromis **n'existe** tout simplement **pas** : on doit parfois conclure que les efforts qui seraient nécessaires pour se protéger contre un risque plausible seraient trop pénibles, et qu'il vaut mieux courir ce risque… ou bien, tout simplement, ne pas utiliser d'outils numériques pour stocker certaines données ou pour parler de certaines choses. D'autres moyens existent, à l'efficacité prouvée de longue date : certains manuscrits de la Bible ont survécu des siècles durant, enfouis dans des jarres entreposées dans des grottes…

7.2 Comment faire ?

Il s'agit de répondre à la question suivante : quel ensemble de pratiques, d'outils me protégeraient de façon suffisante contre les risques évalués précédemment ?

page 49

Vous pouvez par exemple partir de vos pratiques actuelles, et vous mettre dans la peau de l'adversaire — aussi nauséabonde soit-elle — pour vous poser les questions suivantes :

1. Face à une telle politique de sécurité, quels sont les angles d'attaque les plus praticables ?
2. Quels sont les moyens à mettre en œuvre pour ce faire ?
3. Croyez-vous que ces moyens puissent être utilisés par les adversaires ?

Si vous répondez « oui » à la troisième question, prenez le temps de vous renseigner sur les solutions qui permettraient de vous protéger contre ces attaques, puis imaginez les modifications de pratiques entraînées par ces solutions et la politique de sécurité qui en découle. Si ça vous semble praticable, remettez-vous dans la peau de l'adversaire, et posez-vous à nouveau les questions énoncées ci-dessus.

Réitérez ce processus de réflexion, recherche et imagination jusqu'à trouver une voie praticable, un compromis tenable.

En cas d'incertitude, il est toujours possible de demander à une personne digne de confiance et plus compétente en la matière de se mettre dans la peau de l'adversaire : elle sera ravie de constater que vous avez fait vous-mêmes le gros du travail de réflexion, ce qui l'encouragera certainement à vous aider sur les points qui restent hors de votre portée.

7.3 Quelques règles

Avant de s'intéresser de plus près à l'étude de cas concrets et des politiques de sécurité qu'il serait possible de mettre en place, il existe quelques grands principes, quelques grandes familles de choix…

7.3.1 Complexe vs. simple

En matière de sécurité, une solution simple doit toujours être préférée à une solution complexe.

Tout d'abord, parce qu'une solution complexe offre plus de « surface d'attaque », c'est-à-dire plus de lieux où peuvent apparaître des problèmes de sécurité… ce qui ne manquera pas d'arriver.

Ensuite, parce que plus une solution est complexe, plus il faut de connaissances pour l'imaginer, la mettre en œuvre, la maintenir… mais aussi pour l'examiner, évaluer sa pertinence et ses problèmes. Ce qui fait qu'en règle générale, plus une solution est complexe, moins elle aura subi les regards acérés — et extérieurs — nécessaires pour établir sa validité.

Enfin, tout simplement, une solution complexe, qui ne tient pas en entier dans l'espace mental des personnes qui l'ont élaborée, a plus de chances de générer des problèmes de sécurité issus d'interactions complexes ou de cas particuliers difficiles à déceler.

Par exemple, plutôt que de passer des heures à mettre en place des dispositifs visant à protéger un ordinateur particulièrement sensible contre les intrusions provenant du réseau, autant l'en débrancher. On peut même parfois retirer physiquement la carte réseau…

page 13

7.3.2 Liste blanche, liste noire

Le réflexe courant, lorsqu'on prend connaissance d'une menace, est de chercher à s'en prémunir. Par exemple, après avoir découvert que tel logiciel laisse des traces de nos activités dans tel dossier, on nettoiera régulièrement cet emplacement. Jusqu'à découvrir que le même logiciel laisse aussi des traces dans un autre dossier, et ainsi de suite.

C'est le principe de la liste noire [1] : une liste des dossiers où sont enregistrés les fichiers temporaires, de logiciels qui envoient des rapports, *etc.* ; cette liste est complétée au fil des découvertes et des mauvaises surprises ; sur cette base, on essaie de faire au mieux pour se prémunir de chacune de ces menaces. Autrement dit, une liste noire fonctionne sur la base de la *confiance-sauf-dans-certains-cas*.

Le principe de la liste blanche est inverse, car c'est celui de la *méfiance-sauf-dans-certains-cas*. On interdit *tout*, *sauf* ce qu'on autorise explicitement. On interdit l'enregistrement de fichiers sur le disque dur, sauf à tel endroit, à tel moment. On interdit aux logiciels d'accéder au réseau, sauf certains logiciels bien choisis. page 115

Voilà pour les principes de base.

Toute politique de sécurité basée sur le principe de la *liste noire* a un gros problème : une telle liste n'est jamais complète, car elle prend uniquement en compte les problèmes qui ont déjà été repérés. C'est une tâche sans fin, désespérante, que de tenir à jour une liste noire ; qu'on le fasse nous-mêmes ou qu'on le délègue à des gens ayant des connaissances informatiques pointues, quelque chose sera forcément oublié.

L'ennui, c'est que malgré leurs défauts rédhibitoires, les outils basés sur une approche *liste noire* sont légion (comme nous allons le voir), au contraire de ceux s'appuyant sur la méthode *liste blanche*, qui nous est donc, sans doute, moins familière.

Mettre en œuvre l'approche *liste blanche* requiert donc un effort initial qui, s'il peut être important, est bien vite récompensé : apprendre à utiliser un système *live* qui n'écrit rien sur le disque dur sans qu'on lui demande, ça prend un temps non négligeable, mais une fois que c'est fait, c'en est fini des longues séances de nettoyage de disque dur, toujours à recommencer, et inefficaces car basées sur le principe de *liste noire*. page 99

Une autre illustration nous est fournie par les logiciels antivirus, qui visent à empêcher l'exécution de programmes mal intentionnés. Vu qu'ils fonctionnent sur le principe de la liste noire, leurs bases de données doivent perpétuellement être mises à jour, systématiquement en retard. Une réponse à ce problème, avec l'approche liste blanche, est d'empêcher l'exécution de tout programme qui n'a pas été enregistré au préalable, ou de limiter les possibilités d'action de chaque programme ; ces techniques, nommées *Mandatory Access Control*, nécessitent aussi de maintenir des listes, mais il s'agit dans ce cas de listes *blanches*, et le symptôme d'une liste obsolète sera le dysfonctionnement d'un logiciel, plutôt que le piratage de l'ordinateur.

Aussi, il est bien plus intéressant de se donner les moyens, lorsque c'est possible, de s'appuyer sur des listes blanches les plus vastes possible, afin de pouvoir faire plein de choses chouettes avec des ordinateurs, dans une certaine confiance. Et de s'appuyer, quand la liste blanche adéquate n'existe pas, sur des listes noires solides, de provenance connue, en gardant en tête le problème intrinsèque à cette méthode ; listes noires qu'on aidera éventuellement à compléter, en partageant nos découvertes.

1. Les expressions « liste blanche » et « liste noire » peuvent évoquer une dimension raciste, que ce soient les termes en eux-mêmes, ou leur hiérarchisation. Cependant, il nous a semblé peu judicieux de ne pas utiliser les termes consacrés et actuellement utilisés par tous les programmes, modes d'emploi et autres documentations techniques.

7.3.3 On n'est pas des robots

Certaines pratiques très exigeantes peuvent être diablement efficaces… jusqu'à ce qu'on commette une erreur. Alors comme on finira forcément par en faire une, il vaut mieux les prévoir plutôt que de payer les pots cassés.

Par exemple, une clé USB destinée à n'être utilisée que sur des ordinateurs utilisant un système libre, et qu'on fait vraiment attention à ne pas laisser traîner, peut *quand même* finir par être oubliée sur une table… et être branchée sur Windows par une personne qui l'aura confondue avec une autre. Mais si elle a été formatée dès le départ avec un système de fichiers incompatible avec Windows, ça devrait limiter la casse…

page 16

Bref, on n'est pas des robots. Il vaut mieux se donner de solides garde-fous matériels, que de s'imposer une vigilance sans bornes — ça permet aussi de garder l'esprit tranquille.

7.3.4 Date limite de consommation

Une fois une politique de sécurité définie, il ne faut pas oublier de la revoir de temps en temps ! Le monde de la sécurité informatique évolue très vite, et une solution considérée comme raisonnablement sûre à l'heure actuelle peut très bien être aisément attaquable l'an prochain.

N'oublions pas non plus de penser dans nos politiques de sécurité qu'il est important de surveiller la vie des logiciels dont on dépend : leurs problèmes, avec une incidence sur la sécurité, leurs mises à jour, avec parfois de bonnes ou de mauvaises surprises… Tout cela prend un peu de temps, et autant le prévoir dès le départ.

page 169

Cas d'usages

Trêve de théorie, illustrons maintenant ces notions avec quelques *cas d'usage* : à partir de situations données, nous indiquerons des pistes permettant de définir une politique de sécurité adéquate. Bon nombre des solutions techniques retenues seront expliquées dans la partie suivante, vers laquelle nous renverrons au besoin.

page 81

Vu qu'ils s'inscrivent tous dans le contexte hors-connexions de ce premier tome, ces cas d'usage auront quelque chose d'artificiel : ils partent tous du principe que les ordinateurs en jeu ne sont jamais connectés à des réseaux, et en particulier à Internet.

Chapitre 8

Cas d'usage : un nouveau départ, pour ne plus payer les pots cassés

(ou comment faire le ménage sur un ordinateur après des années de pratiques insouciantes)

8.1 Contexte

Prenons un ordinateur utilisé sans précautions particulières pendant plusieurs années. Cette machine pose sans doute un ou plusieurs des problèmes suivants :

1. son disque dur conserve des traces indésirables du passé ; page 19
2. le système d'exploitation est un logiciel propriétaire (exemple : Windows), et truffé de logiciels malveillants. page 23

Par ailleurs, des fichiers gênants y sont stockés de façon parfaitement transparente. En effet, cet ordinateur est utilisé pour diverses activités populaires, parmi lesquelles certaines, osons l'avouer, sont parfaitement légales, telles que :

- écouter de la musique et regarder des films pris sur Internet ;
- aider des sans-papiers à préparer leurs dossiers pour la préfecture ;
- dessiner une jolie carte de vœux pour Mamie ;
- fabriquer de menus faux papiers simplifiant grandement les démarches administratives (gonfler des fiches de paie, quand on en a marre de se voir refuser des locations, appart' après appart') ;
- tenir à jour la comptabilité familiale ;
- fabriquer des textes, musiques ou vidéos « terroristes » — plus précisemment menaçant, selon la définition européenne du terrorisme[1], « *de causer […] des destructions massives […] à une infrastructure […] susceptible […] de produire des pertes économiques considérables* », « *dans le but de […] contraindre indûment des pouvoirs publics […] à accomplir ou à s'abstenir d'accomplir un acte quelconque* » ; par exemple, des employés de France Télécom qui, lors d'une lutte, menaceraient de mettre hors d'état de nuire le système de facturation, et d'ainsi permettre aux usagers de téléphoner gratuitement.

8.2 Évaluer les risques

8.2.1 Que veut-on protéger ?

Appliquons au cas présent les catégories définies lorsque nous parlions d'évaluation des risques : page 49

1. EUR-lex, 2002, *Décision-cadre 2002/475/JAI du Conseil de l'Union Européenne, relative à la lutte contre le terrorisme* [http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32002F0475:FR:NOT].

- confidentialité : éviter qu'un œil indésirable ne tombe trop aisément sur les informations stockées dans l'ordinateur ;
- intégrité : éviter que ces informations ne soient modifiées à notre insu ;
- accessibilité : faire en sorte que ces informations restent accessibles quand on en a besoin.

Ici, accessibilité et confidentialité sont prioritaires.

8.2.2 Contre qui veut-on se protéger ?

Cette question est importante : en fonction de la réponse qu'on lui donne, la politique de sécurité adéquate peut varier du tout au tout.

Geste généreux, conséquences judiciaires

Cet ordinateur pourrait être saisi lors d'une perquisition.

Par exemple, votre fils a généreusement donné un gramme de *shit* à un ami fauché, qui, après s'être fait pincer, a informé la police de la provenance de la chose… à la suite de quoi votre fils est pénalement considéré comme trafiquant de stupéfiants. D'où la perquisition.

Dans ce genre de cas, l'ordinateur a de grandes chances d'être examiné par la police, mettant en péril l'objectif de confidentialité. La gamme de moyens qui seront probablement mis en œuvre va du gendarme de Saint-Tropez, allumant l'ordinateur et cliquant partout, à l'expert judiciaire qui examinera de beaucoup plus près le disque dur ; il est en revanche improbable que des moyens extra-légaux, usuellement aux mains des services spéciaux et des militaires, soient utilisés dans cette affaire.

Cambriolage

Cet ordinateur pourrait être dérobé lors d'un cambriolage.

Au contraire de la police, les voleurs n'ont sans doute pas grand-chose à faire de vos petits secrets… et ne vous dénonceront pas. Au pire vous feront-ils chanter à propos de la récupération de vos données. Il est cependant improbable qu'ils mettent en œuvre de grands moyens pour les retrouver sur le disque dur de l'ordinateur.

8.3 Définir une politique de sécurité

Posez-vous maintenant, en vous mettant dans la peau de l'adversaire, les questions exposées dans notre méthodologie.

page 51

8.3.1 Première étape : quand ouvrir les yeux suffit pour voir

1. Angle d'attaque le plus praticable : brancher le disque dur sur un autre ordinateur, examiner son contenu, y trouver tous vos petits secrets.
2. Moyens nécessaires : un autre ordinateur, dont le gendarme de Saint-Tropez se servira pour trouver le plus gros de vos secrets ; un expert judiciaire, lui, saurait aussi retrouver les fichiers que vous croyiez avoir effacés ; Nostradamus en déduirait la date de levée de vos semis.
3. Crédibilité de l'attaque : grande.

page 37

Il faut donc adapter vos pratiques. Contre ce type d'attaque, chiffrer le disque dur est la réponse évidente : installer et utiliser un système chiffré est désormais relativement simple.

page 105

Les étapes pour y arriver seraient alors :

1. Lancer un système *live* afin d'effectuer les opérations suivantes dans un contexte relativement sûr :
   - sauvegarder temporairement, sur un disque externe ou une clé USB chiffrés, les fichiers qui doivent survivre au grand nettoyage ;
   - éjecter/démonter et débrancher ce support de stockage externe ;
   - effacer « pour de vrai » l'intégralité du disque dur **interne** de l'ordinateur.
2. Installer un système d'exploitation libre, en précisant au programme d'installation de chiffrer le disque dur, mémoire virtuelle (*swap*) comprise. page 17
3. Recopier vers le nouveau système les données préalablement sauvegardées.
4. Mettre en place ce qu'il faut pour supprimer des fichiers de façon « sécurisée », afin de pouvoir…
5. Effacer le contenu des fichiers qui se trouvent sur le support de sauvegarde temporaire, qui pourra éventuellement resservir.

Et ensuite, de temps à autre, faire en sorte que les données supprimées sans précautions particulières ne soient pas récupérables par la suite. Il faudra également veiller à mettre régulièrement à jour le système, afin de combler les « trous de sécurité » que pourraient utiliser des logiciels malveillants. page 24

Pour effectuer ces étapes, se référer aux recettes suivantes :

- chiffrer un disque externe ou une clé USB, voir page 137 ;
- utiliser un système *live*, voir page 99 ;
- sauvegarder des données, voir page 143 ;
- effacer « pour de vrai », voir page 127 ;
- installer un système chiffré, voir page 105 ;
- garder un système à jour, voir page 169.

Cette voie semblant praticable, posons-nous, de nouveau, les mêmes questions.

8.3.2 Seconde étape : le tiroir de la commode n'était pas chiffré

1. Angle d'attaque : l'équivalent des fichiers qu'on cherche à protéger traîne peut-être dans la pièce voisine, dans le troisième tiroir de la commode, sur papier ou sur une clé USB.
2. Moyens nécessaires : perquisition, cambriolage, ou autre visite impromptue.
3. Crédibilité de l'attaque : grande, c'est précisément contre ce type de situations qu'on cherche à se protéger ici.

Là encore, on constate qu'une politique de sécurité doit être pensée comme un tout. Sans un minimum de cohérence dans les pratiques, rien ne sert de s'embêter à taper des phrases de passe longues comme un jour sans pain. page 51

Il est donc temps de trier les papiers dans la commode, et de nettoyer toute clé USB, CD, DVD contenant des données qu'on compte désormais chiffrer :

- sauvegarder sur un support chiffré les données à conserver
- pour les clés USB et disques durs externes : effacer pour de vrai leur contenu ; page 127
- pour les CD et DVD : les détruire, et se débarrasser des résidus ;
- décider que faire des données préalablement sauvegardées : les recopier sur le disque dur nouvellement chiffré ou les archiver. page 77

8.3.3 Troisième étape : la loi comme moyen de coercition

1. Angle d'attaque : la police a le droit d'exiger que vous lui donniez accès aux informations chiffrées, comme expliqué dans le chapitre consacré à la cryptographie. page 37
2. Moyens nécessaires : suffisamment de persévérance dans l'enquête pour appliquer cette loi.

3. Crédibilité de l'attaque : encore faut-il que la police considère pouvoir trouver des éléments à charge sur l'ordinateur, avec suffisamment de foi pour pousser le bouchon jusque-là. Dans le strict cadre de l'enquête qui part du gramme de *shit*, c'est peu probable, mais pas du tout impossible.

Si la police en arrive à exiger l'accès aux données chiffrées, se posera, en pratique, la question suivante : les informations contenues dans l'ordinateur font-elles encourir plus de risques que le refus de donner la phrase de passe ? Après, c'est selon comment on le sent. Céder, dans cette situation, ne remet pas en cause tout l'intérêt de chiffrer, au départ, son disque dur : ça permet tout au moins de savoir ce qui a été dévoilé, quand, et à qui.

Ceci dit, il peut être bon de s'organiser pour vivre de façon moins délicate une telle situation : le nouvel objectif pourrait être d'avoir un disque dur suffisamment « propre » pour que ce ne soit pas la catastrophe si on cède face à la loi, ou si le système cryptographique utilisé est cassé.

Comme premier pas, il est souvent possible de faire un compromis concernant l'accessibilité, pour des fichiers concernant des projets achevés dont on n'aura pas besoin souvent ; on traitera ceci dans le cas d'usage sur l'archivage, qu'il pourra être bon d'étudier après celui-ci.

page 77

Ensuite, c'est toute la question de la compartimentation qui se pose ; en effet, s'il est possible d'augmenter globalement, de nouveau, le niveau de sécurité de l'ensemble des activités pratiquées… ce serait trop pénible à l'usage. Il convient donc de préciser les besoins respectifs, en termes de confidentialité, de ces diverses activités. Et, à partir de là, faire le tri et décider lesquelles, plus « sensibles » que les autres, doivent bénéficier d'un traitement de faveur.

Le prochain cas d'usage étudiera de tels traitements de faveur, mais patience, mieux vaut pour l'instant terminer la lecture de celui-ci !

page 65

8.3.4 Quatrième étape : en réseau

Tout ceci est valable pour un ordinateur hors-ligne. D'autres angles d'attaques sont imaginables, s'il est connecté à un réseau. Le second tome de ce guide les étudiera.

tome 2 § 3.4

Et au-delà de ces problèmes, plusieurs autres angles d'attaque demeurent encore envisageables contre une telle politique de sécurité.

8.3.5 Angle d'attaque : une brèche dans le système de chiffrement utilisé

Comme il a déjà été expliqué en ces pages, tout système de sécurité finit par être cassé. Si l'algorithme de chiffrement utilisé est cassé, ça fera la une des journaux, tout le monde sera au courant, et il sera possible de réagir.

Mais si c'est sa mise en œuvre dans le noyau Linux qui est cassée, ça ne passera pas dans *Libération*, et il y a fort à parier que seuls les spécialistes de la sécurité informatique seront au courant.

Lorsqu'on ne côtoie pas de tels êtres, une façon de se tenir au courant est de s'abonner aux annonces de sécurité de Debian [2]. Les emails reçus par ce biais sont rédigés en anglais, mais ils donnent l'adresse de la page où on peut trouver leur traduction française. La difficulté, ensuite, est de les interpréter…

2. La liste de diffusion se nomme debian-security-announce [http://lists.debian.org/debian-security-announce/].

Ceci étant dit, même si le système de chiffrement utilisé est « cassé », encore faut-il que les adversaires le sachent… le gendarme de Saint-Tropez n'en saura rien, mais un expert judiciaire, si.

Par ailleurs, dans le rayon science-fiction, rappelons qu'il est difficile de connaître l'avance qu'ont, en la matière, militaires et agences gouvernementales — comme la NSA.

8.3.6 Angle d'attaque : *cold boot attack*

1. Angle d'attaque : la *cold boot attack* est décrite dans le chapitre consacré aux traces. page 19
2. Moyens nécessaires : accéder physiquement à l'ordinateur pendant qu'il est allumé ou éteint depuis peu, par exemple lors d'une perquisition.
3. Crédibilité de l'attaque : à notre connaissance, cette attaque n'a jamais été utilisée, du moins de façon publique, par des autorités. Sa crédibilité est donc très faible.

Il peut sembler superflu de se protéger contre cette attaque dans la situation décrite ici, mais mieux vaut prendre, dès maintenant, de bonnes habitudes, plutôt que d'avoir de mauvaises surprises dans quelques années. Quelles habitudes ? En voici quelques-unes qui rendent plus difficile cette attaque :

- éteindre l'ordinateur lorsqu'on ne s'en sert pas ;
- prévoir la possibilité de couper le courant facilement et rapidement : interrupteur de multiprise aisément accessible, ôter la batterie d'un ordinateur portable quand il est branché sur le secteur (… il suffit alors de débrancher le cordon secteur pour éteindre la machine) ;
- rendre l'accès au compartiment de votre ordinateur contenant la RAM plus long et difficile, par exemple en le collant/soudant.

8.3.7 Angle d'attaque : l'œil et la vidéo-surveillance

Avec le système chiffré imaginé à la première étape, la confidentialité des données repose sur le fait que la phrase de passe soit gardée secrète. Si elle est tapée devant une caméra de vidéo-surveillance, un adversaire ayant accès à cette caméra ou à ses éventuels enregistrements pourra découvrir ce secret, puis se saisir de l'ordinateur et avoir accès aux données. Plus simplement, un œil attentif, dans un bar, pourrait voir la phrase de passe pendant qu'elle est tapée.

Monter une telle attaque nécessite de surveiller les personnes utilisant cet ordinateur, jusqu'à ce que l'une d'entre elles tape la phrase de passe au mauvais endroit. Ça peut prendre du temps et c'est coûteux.

Dans la situation décrite ici, une telle attaque relève de la pure science-fiction ; à l'heure actuelle, rares sont les organisations susceptibles de mettre en œuvre des moyens aussi conséquents, mis à part divers services spéciaux : anti-terroristes, espionnage industriel…

Pour se prémunir d'une telle attaque, il convient de :

- choisir une longue phrase de passe, qui rend impossible la mémorisation « à la volée » par un observateur humain ; page 91
- vérifier autour de soi, à la recherche d'éventuels yeux (humains ou électroniques) indésirables, avant de taper sa phrase de passe.

8.3.8 Angle d'attaque : la partie non-chiffrée et le micrologiciel

Comme expliqué dans la recette dédiée, un système « chiffré » ne l'est pas entière- page 105

ment : le petit logiciel qui nous demande, au démarrage, la phrase de passe de chiffrement du *reste* des données, est, lui, stocké en clair sur la partie du disque dur qu'on nomme `/boot`. Un attaquant ayant accès à l'ordinateur peut aisément, en quelques minutes, modifier ce logiciel, y installer un *keylogger*, qui conservera la phrase de passe, pour venir la chercher plus tard, ou, tout simplement, l'enverra par le réseau.

page 23

Si cette attaque est montée à l'avance, l'adversaire pourra déchiffrer le disque dur quand il se saisira de l'ordinateur, lors d'une perquisition par exemple.

Les moyens nécessaires pour cette attaque sont, somme toute, assez limités : *a priori*, point n'est besoin d'être Superman pour avoir accès, pendant quelques minutes, à la pièce où réside l'ordinateur.

Cependant, là aussi, dans la situation décrite pour ce cas d'usage, nous sommes en pleine science-fiction. Mais la réalité a parfois tendance à dépasser la fiction…

> La seule protection praticable contre cette attaque est de stocker les programmes de démarrage, dont ce petit dossier non-chiffré (`/boot`), sur un support externe, comme une clé USB, qui sera conservé en permanence dans un endroit plus sûr que l'ordinateur. C'est l'*intégrité* de ces données, et non leur *confidentialité*, qui est alors à protéger. Cette pratique exige pas mal de compétences et de rigueur ; nous ne la développerons pas dans ce guide.
>
> De telles pratiques mettent la barre plus haut, mais il reste un mais : une fois obtenu l'accès physique à l'ordinateur, si `/boot` n'est pas accessible, et donc pas modifiable, il reste possible d'effectuer le même type d'attaque sur le micrologiciel de la machine. C'est légèrement plus difficile, car la façon de faire dépend du modèle d'ordinateur utilisé, mais c'est possible. Nous ne connaissons aucune façon praticable de s'en protéger.

8.3.9 Angle d'attaque : les logiciels malveillants

page 23

Nous avons appris dans un chapitre précédent que des logiciels installés à notre insu sur un ordinateur peuvent nous dérober des données. Dans le cas présent, un tel logiciel est en mesure de transmettre la clé de chiffrement du disque dur à un adversaire… qui obtiendra ensuite, grâce à cette clé, l'accès aux données chiffrées, quand il aura accès physique à l'ordinateur.

Installer un logiciel malveillant sur le système Debian dont il est question ici requiert des compétences de plus haut niveau que les attaques étudiées ci-dessus, mais aussi plus de préparation. Une telle attaque relève donc, ici aussi, de la science-fiction, du moins en ce qui concerne la situation qui nous occupe. Dans d'autres situations, il conviendra parfois d'être extrêmement prudent quant à la provenance des données et logiciels qu'on injecte dans l'ordinateur, en particulier lorsqu'il est connecté à Internet… un cas qui, rappelons-le, n'est pas notre propos dans ce premier tome.

page 117

La recette concernant l'installation de logiciels donne quelques pistes fort utiles sur la façon d'installer de nouveaux logiciels proprement. Le second tome de ce guide, consacré aux réseaux, et à Internet en particulier, prolonge cette étude.

8.3.10 Angle d'attaque : la force brute

Attaquer un système cryptographique par « force brute » est la plus simple, la plus stupide, et la plus lente des manières. Mais quand on ne peut mettre en œuvre un autre type d'attaque…

Pour le disque dur chiffré lors de l'étape 1, ça demande énormément de temps (de nombreuses années) et/ou énormément d'argent et des compétences pointues… du moins si la phrase de passe est solide.

Ce qu'on peut se dire, c'est qu'*a priori*, si une organisation est prête à mobiliser autant de ressources pour avoir accès à vos données, elle gagnerait amplement à mettre en place une des autres attaques, moins coûteuses et tout aussi efficaces, listées ci-dessus.

Chapitre 9

Cas d'usage : travailler sur un document sensible

9.1 Contexte

Après avoir pris un nouveau départ, l'ordinateur utilisé pour mener ce projet à bien a été équipé d'un système chiffré. Bien. Survient alors le besoin de travailler sur un projet particulier, plus « sensible », par exemple :

page 57
page 105

- un tract doit être rédigé ;
- une affiche doit être dessinée ;
- un livre doit être maquetté puis exporté en PDF ;
- une fuite d'informations doit être organisée pour divulguer les affreuses pratiques d'un employeur ;
- un film doit être monté et gravé sur DVD.

Dans tous ces cas, les problèmes à résoudre sont à peu près les mêmes.

Comme il serait trop pénible d'augmenter globalement, de nouveau, le niveau de sécurité de l'ordinateur, il est décidé que ce projet particulier doit bénéficier d'un traitement de faveur.

9.1.1 Conventions de vocabulaire

Par la suite, nous nommerons :

- les *fichiers de travail* : l'ensemble des fichiers nécessaires à la réalisation de l'œuvre : les images ou *rushes* utilisés comme bases, les documents enregistrés par le logiciel utilisé, *etc.* ;
- l'*œuvre* : le résultat final (tract, affiche, *etc.*)

En somme, la matière première, et le produit fini.

9.2 Évaluer les risques

Partant de ce contexte, tentons maitenant de définir les risques auxquels exposent les pratiques décrites dans ce cas d'usage.

9.2.1 Que veut-on protéger ?

Appliquons au cas présent les catégories définies lorsque nous parlions d'évaluation des risques :

page 49

- confidentialité : éviter qu'un œil indésirable ne découvre trop aisément l'œuvre et/ou les fichiers de travail ;

- intégrité : éviter que ces documents ne soient modifiés à notre insu ;
- accessibilité : faire en sorte que ces documents restent accessibles quand on en a besoin.

Ici, accessibilité et confidentialité sont prioritaires.

Accessibilité, car l'objectif principal est tout de même de réaliser l'œuvre. S'il fallait se rendre au pôle Nord pour ce faire, le projet risquerait fort de tomber à l'eau.

Et pour ce qui est de la confidentialité, tout dépend de la publicité de l'œuvre. Voyons donc ça de plus près.

Œuvre à diffusion restreinte

Si le contenu de l'œuvre n'est pas complètement public, voire parfaitement secret, il s'agit de dissimuler à la fois l'œuvre *et* les fichiers de travail.

Œuvre diffusée publiquement

Si l'œuvre a vocation à être publiée, la question de la confidentialité se ramène à celle de l'anonymat.

C'est alors, principalement, les fichiers de travail qui devront passer sous le tapis : en effet, les découvrir sur un ordinateur incite fortement à penser que ses propriétaires ont réalisé l'œuvre… avec les conséquences potentiellement désagréables que cela peut avoir.

Mais ce n'est pas tout : si l'œuvre, ou ses versions intermédiaires, sont stockées sur cet ordinateur (PDF, *etc.*), leur date de création est très probablement enregistrée dans le système de fichiers et dans des méta-données. Le fait que cette date soit antérieure à la publication de l'œuvre peut aisément amener des adversaires à tirer des conclusions gênantes quant à sa généalogie.

page 16
page 21

9.2.2 Contre qui veut-on se protéger ?

Pour faire simple, reprenons les possibilités décrites dans le cas d'usage « un nouveau départ » : l'ordinateur utilisé pour réaliser l'œuvre peut être dérobé, plus ou moins fortuitement, par de quelconques flics, voire par de braves voleurs travaillant à leur compte.

page 57

9.3 Accro à Windows ?

La première question qui se pose est : quel système d'exploitation utiliser ? Ça dépend, évidemment, des logiciels utilisés pour ce projet :

S'ils fonctionnent sous GNU/Linux, continuons la lecture de ce chapitre pour étudier les options qui s'offrent à nous.

S'ils fonctionnent exclusivement sous Windows, c'est dommage. Mais nous étudions tout de même un chemin praticable qui permet de limiter la casse. Allons donc voir à quoi ressemble ce chemin, en ignorant les paragraphes suivants, qui sont consacrés à GNU/Linux.

page 68

9.4 Le système *live* amnésique

Les problèmes attenants à la situation de départ sont les mêmes que ceux du cas d'usage « un nouveau départ ». Mais avant de mettre sur la table de potentielles politiques de sécurité, lançons-nous dans un rapide tour d'horizon des outils et méthodes disponibles.

page 57

9.4.1 Liste noire vs. liste blanche

Vu qu'on a déjà un système Debian chiffré, on peut, de prime abord, imaginer le configurer finement pour qu'il conserve moins de traces de nos activités sur le disque dur. Le problème de cette approche, c'est qu'elle est de type « liste noire », et nous en avons expliqué les limites en ces pages : quel que soit le temps consacré, quelle que soit l'expertise mise au travail, même avec une compréhension particulièrement poussée des entrailles du système d'exploitation utilisé, on oubliera toujours une petite option bien cachée, il restera toujours des traces indésirables auxquelles on n'avait pas pensé.

page 53

Au contraire, certains systèmes *live* fonctionnent sur le principe de la « liste blanche » : tant qu'on ne le demande pas explicitement, aucune trace n'est laissée sur le disque dur.

page 99

En envisageant uniquement le critère « confidentialité », le système *live* bat donc l'autre à plate couture. En termes de temps et de difficulté de mise en œuvre, en revanche, la comparaison est plus mitigée.

9.4.2 Le beurre, ou l'argent du beurre ?

Un système *live* est en effet amnésique ; c'est certes son principal atout, mais cette propriété est aussi source d'inconvénients. Par exemple, dans le cas où notre système *live* préféré ne fournit pas un logiciel donné, qui est pourtant indispensable au projet, il faut, au choix :

- installer le logiciel dans le système *live* au début de chaque session de travail ;
- créer une clé *live* incluant notre logiciel dans son volume persistant ;
- faire du lobbying auprès des auteurs du système *live* pour qu'ils y ajoutent le logiciel souhaité ;

L'utilisation d'un système live est la solution la plus sûre et, dans ce cas, la moins difficile à mettre en place. Auquel cas, allons étudier une politique de sécurité basée là-dessus.

page suivante

À noter qu'il est possible d'installer une Debian dans VirtualBox, mais cette solution est réservée aux utilisateurs avertis, et ne sera donc pas documentée ici.

9.5 Travailler sur un document sensible… sur un système *live*

page 65 Après avoir présenté le contexte dans le début de ce cas d'usage, et avoir décidé d'utiliser un système *live*, reste à mettre cette solution en place… et à étudier ses limites.

9.5.1 Télécharger et installer le système *live*

Tous les systèmes *live* ne sont pas particulièrement destinés à des pratiques « sensibles ». Il importe donc de choisir un système spécialement conçu pour (tenter de) ne laisser aucune trace sur le disque dur de l'ordinateur sur lequel il est utilisé.

Si l'on ne dispose pas encore d'une copie de la dernière version du système live *Tails*,
page 100 suivre la recette télécharger et installer un système *live* « discret ».

page 93 À partir du premier périphérique *Tails* ainsi créé, nous allons créer une clé USB dédiée à notre projet. Pour cela démarrer le système *live* précédemment installé puis suivre
page 101 les recettes cloner une clé Tails puis créer et configurer un volume persistant dans Tails en activant uniquement l'option *Données personnelles*.
page 102

9.5.2 Installer un éventuel logiciel additionnel

Si l'on a besoin d'utiliser un logiciel qui n'est pas installé dans *Tails* et que l'on ne
page 103 veut pas le réinstaller à chaque fois, suivre la recette installer un logiciel additionnel persistant dans Tails.

9.5.3 Utiliser le système live

page 93 Chaque fois que l'on souhaite travailler sur notre document, il suffira de démarrer sur la clé contenant notre système *live* et sa persistance chiffrée puis d'activer le volume
page 102 persistant.

9.5.4 Supprimer le système live

tome 2 ch. 9 Une fois notre projet terminé et imprimé ou publié en ligne, il nous faut supprimer le volume persistant après avoir éventuellement archivé le projet
page 102

9.5.5 Limites

page 77 Certaines limites, communes à cette méthode et à celle basée sur l'usage de Windows,
plus bas sont exposées plus loin.
page 75

9.6 Travailler sur un document sensible… sous Windows

page 65 Après avoir présenté le contexte dans le début de ce cas d'usage et décidé, malgré tous les problèmes que ça pose, d'utiliser Windows, essayons maintenant de trouver une façon de limiter quelque peu la casse.

9.6.1 Point de départ : une passoire et une boîte de rustines desséchées

Partons d'un ordinateur muni, de la façon la plus classique qui soit, d'un disque dur sur lequel Windows est installé. Nous ne nous appesantirons pas sur cette situation, la première partie de cet ouvrage ayant abondamment décrit les multiples problèmes qu'elle pose. Une passoire, en somme, pleine de trous de sécurité.

On peut donc imaginer coller quelques rustines sur cette passoire. Faisons-en rapidement le tour.

Un disque dur, ça se démonte et ça se cache. Certes. Mais il y a les périodes où l'on s'en sert, parfois plusieurs jours ou semaines d'affilée. Cette rustine est basée sur deux hypothèses quelque peu osées :

- *Nous avons de la chance.* Il suffit en effet que l'accident (perquisition, cambriolage, *etc.*) survienne au mauvais moment pour que toute la confidentialité désirée soit réduite à néant ;
- *Notre discipline est parfaitement rigoureuse.* En effet, si l'on oublie, ou qu'on ne prend pas le temps, d'aller « ranger » le disque dur quand on n'en a plus besoin, et que l'accident survient à ce moment-là, c'est perdu, fin de la partie.

Par ailleurs, des outils existent pour chiffrer des données sous Windows. Quelle que soit la confiance qu'on leur accorde, il n'en reste pas moins qu'ils s'appuient obligatoirement sur les fonctions offertes par la boîte noire qu'est Windows. On ne peut donc que s'en méfier, et dans tous les cas, Windows, lui, aura accès à nos données *en clair*, et personne ne sait ce qu'il pourrait bien en faire.

Pour conclure ce petit tour dans la cour des miracles douteux, ajoutons que la seule « solution » possible dans le cas présent serait une approche de type liste noire, dont l'inefficacité crasse a déjà été expliquée précédement. page 51

Il est maintenant temps de passer aux choses sérieuses.

9.6.2 Seconde étape : enfermer Windows dans un compartiment (presque) étanche

Ce qui commence à ressembler à une solution sérieuse, ce serait de faire fonctionner Windows dans un compartiment étanche, dans lequel on ouvrirait, quand c'est nécessaire et en connaissance de cause, une porte pour lui permettre de communiquer avec l'extérieur de façon strictement limitée.

En d'autres termes, mettre en place une solution basée sur une logique de type *liste blanche* : rien ne pourrait entrer dans Windows ou en sortir *a priori*, et à partir de cette règle générale, on autorise des *exceptions*, au cas par cas, en réfléchissant à leur impact.

La *virtualisation*[1] permet de mettre en place ce type de systèmes. C'est un ensemble de techniques matérielles et logicielles qui permettent de faire fonctionner, sur un seul ordinateur, plusieurs systèmes d'exploitation, séparément les uns des autres, (presque) comme s'ils fonctionnaient sur des machines physiques distinctes.

Il est ainsi relativement facile, de nos jours, de faire fonctionner Windows **à l'intérieur** d'un système GNU/Linux, en lui coupant, par la même occasion, tout accès au réseau — et en particulier, en l'isolant d'Internet.

Attention : il est conseillé de lire l'intégralité de ce chapitre **avant** de se précipiter sur les recettes pratiques ; la description de l'hypothèse qui suit est assez longue, et ses limites sont étudiées à la fin de ce chapitre, où des contre-mesures sont envisagées. Il serait quelque peu dommage de passer quatre heures à suivre ces recettes, avant de se rendre compte qu'une toute autre solution serait, en fait, plus adéquate.

Commençons par résumer l'hypothèse proposée.

L'idée est donc de faire fonctionner Windows dans un compartiment *a priori* étanche, **à l'intérieur** d'un système Debian chiffré tel que celui qui a pu être mis en place à la suite de la lecture du cas d'usage précédent. Ce qui servira de disque dur à Windows, c'est en fait un gros fichier, rangé à côté de tous nos autres fichiers, sur le disque dur de notre système Debian chiffré. Ce fichier, qui n'a vraiment rien de particulier, nous le nommons une *image de disque virtuel*, parfois abrégé par une *image de disque*. page 57

1. Pour plus d'informations, voir la page Wikipédia, 2014, *Virtualisation* [https://fr.wikipedia.org/wiki/Virtualisation].

Le fait que ce pseudo-disque dur soit un fichier nous simplifiera grandement la vie par la suite, qui décrit plus précisément la procédure envisagée.

Installer VirtualBox

page 156 La recette « installer Virtual Box » explique comment installer le logiciel VirtualBox, qui nous servira à lancer Windows dans un compartiment étanche.

Installer un Windows « propre » dans VirtualBox

page 157 Préparons une image de disque virtuel *propre* : la recette « installer un Windows virtualisé » explique comment installer Windows dans VirtualBox en lui coupant, **dès le départ**, tout accès au réseau.

À partir de ce moment-là, on qualifie Windows de système *invité* par le système Debian chiffré, qui, lui, est le système *hôte*.

Installer les logiciels nécessaires dans le Windows « propre »

Autant installer, dès maintenant, dans le Windows « propre », tout logiciel *non compromettant*[2] nécessaire à la réalisation des œuvres préméditées : ça évitera de le refaire au début de chaque nouveau projet… et ça évitera, souhaitons-le ardemment, d'utiliser une image Windows « sale » pour un nouveau projet, un jour où le temps presse.

Vu que le Windows *invité* n'a pas le droit de sortir de sa boîte pour aller chercher lui-même des fichiers, il est nécessaire de lui faire parvenir depuis « l'extérieur » les fichiers d'installation des logiciels nécessaires.

Une telle opération sera aussi utile, par la suite, pour lui envoyer toutes sortes de fichiers, et nous y reviendrons. Pour l'heure, vu que nous sommes en train de préparer une image de Windows « propre », servant de base à chaque nouveau projet, ne mélangeons pas tout, et contentons-nous de lui envoyer uniquement ce qui est nécessaire à l'installation des logiciels non compromettants souhaités.

Créons, sur le système hôte, un dossier nommé *Logiciels Windows*, et copions-y **uniquement** les fichiers nécessaires à l'installation des logiciels souhaités.

page 164 Puis partageons ce dossier avec le Windows *invité*, sans rendre ce partage permanent ; la recette « envoyer des fichiers au système virtualisé » explique comment procéder pratiquement.

Et en ce qui concerne l'installation des logiciels à l'intérieur du Windows *invité* : toute personne suffisamment accro à Windows pour lire ces pages est, sans aucun doute, plus compétente que celles qui écrivent ces lignes.

Attention : une fois cette étape effectuée, il est impératif de ne **rien** faire d'autre dans ce Windows virtualisé.

Congeler le Windows « propre »

Congelons maintenant l'image de disque *propre* qui vient d'être préparée, c'est-à-dire : sauvegardons-la dans un coin, telle quelle, et on ne démarrera plus **jamais** dessus. Par la suite, elle ne servira plus que de base de départ.

page 160 La recette sauvegarder une image de disque virtuel explique comment effectuer cette opération.

2. S'il est nécessaire de cacher qu'on fabrique des films, avoir des logiciels de montage vidéo peut être compromettant, parce qu'il serait plus difficile de nier cette activité, si cela s'avérait nécessaire.

Nouveau projet, nouveau départ

Mettons qu'un nouveau projet nécessitant l'utilisation de Windows débute ; voici ce qui se passe :

1. l'image de disque propre est clonée, pour donner naissance à une nouvelle image de disque, en tout point identique ; c'est la *décongélation* ;
2. la nouvelle image de disque, issue de la décongélation, peut maintenant être démarrée dans son compartiment étanche ; elle servira **exclusivement** pour le nouveau projet, et devient désormais une image *sale* ;
3. au sein de cette nouvelle image *sale*, un nouvel utilisateur Windows est créé ; le nom qui lui est attribué doit être différent **à chaque fois** qu'un nouveau projet est ainsi démarré, et cet utilisateur servira **exclusivement** pour ce nouveau projet. Ceci, parce que les logiciels tendent à inscrire le nom de l'utilisateur actif dans les méta-données des fichiers qu'ils enregistrent, et qu'il vaut mieux éviter de rendre possibles de fâcheux recoupements.

page 21

La recette créer une nouvelle machine virtuelle à partir d'une image propre explique les détails techniques de la chose.

page 162

Maintenant que nous avons un compartiment étanche, voyons comment y ouvrir des portes sélectivement, en fonction des besoins.

Comment envoyer des fichiers au Windows embastillé ? Vu que le Windows *invité* n'a pas le droit de sortir de sa boîte pour aller chercher lui-même des fichiers, il peut être nécessaire de lui en faire parvenir depuis « l'extérieur », par exemple :

- de la matière première (*rushes*, images ou textes provenant d'autres sources) ;
- un logiciel nécessaire au nouveau projet, et absent de l'image virtuelle *décongelée*.

Nous avons déjà vu comment procéder, mais c'était dans un cas très particulier : l'installation de nouveaux logiciels dans un Windows « propre » *invité*. Partager des fichiers avec un Windows « sale » requiert davantage de réflexion et de précautions, que nous allons maintenant étudier.

La façon de faire est légèrement différente, en fonction du support sur lequel se trouvent, à l'origine, les fichiers à importer (CD, DVD, clé USB, dossier présent sur le disque dur du système chiffré), mais les précautions d'usage sont les mêmes :

- Windows doit **uniquement** avoir accès aux fichiers qu'on veut y importer, et c'est tout. Il n'est pas question de lui donner accès à un dossier qui contient, pêle-mêle, des fichiers concernant des projets qui ne devraient pas être recoupés entre eux. Si ça implique de commencer par une phase de tri et de rangement, et bien, soit.
- Lorsque Windows a besoin de *lire* (recopier) les fichiers contenus dans un dossier, on lui donne **uniquement** accès en *lecture* à ce dossier. Le moins on donne le droit à Windows d'écrire ici ou là, le moins il laissera de traces gênantes.

À noter que, lorsqu'on décide de partager un dossier du système hôte avec un Windows *invité*, VirtualBox propose de rendre ce partage permanent. Ça évite de refaire la manipulation à chaque fois qu'il est nécessaire d'envoyer un fichier au Windows *invité*, mais ça implique le risque de déposer des fichiers dans ce dossier sans penser qu'ils pourront être lisibles par Windows et ses sbires.

C'est pourquoi, afin d'éviter de se mélanger les pinceaux, nous recommandons de :

- créer **un** dossier d'importation par projet ;
- nommer ce dossier de façon aussi explicite que possible ; par exemple : *Dossier lisible par Windows* ;
- ne jamais partager d'autres dossiers que celui-ci avec le Windows *invité*.

La recette « envoyer des fichiers au système virtualisé » explique comment procéder pratiquement.

page 164

Comment faire sortir des fichiers du Windows embastillé ? Le Windows *invité* n'a pas le droit, par défaut, de laisser des traces en dehors de son compartiment étanche. Mais presque inévitablement vient le temps où il est nécessaire d'en faire sortir des fichiers, et à ce moment-là, il nous faut l'autoriser explicitement, par exemple :

- pour emmener à la boîte-à-copies, ou chez l'imprimeur, un fichier PDF exporté ;
- pour projeter, sous forme de DVD, le film fraîchement réalisé.

Lorsqu'on doit récupérer un CD ou DVD non chiffré, et que la machine hôte est munie d'un graveur, il suffit de « prêter » ce périphérique, temporairement, au Windows *invité*, afin de graver depuis ce système.

Dans le cas où rien n'oblige à récupérer les fichiers sur un support non chiffré, il est possible de les exporter vers un dossier vide, dédié à cet usage, et stocké sur un volume chiffré qui peut être :

- une clé USB chiffrée, qu'on active sous Debian en tapant la phrase de passe correspondante ;
- le disque dur de la Debian chiffrée qui fait ici office de système *hôte*.

Ce dossier dédié sera partagé, via VirtualBox, avec le Windows *invité*. Insistons sur les mots **vide** et **dédié** : Windows pourra lire et modifier tout ce que ce dossier contient, et il serait dommageable de lui permettre de lire des fichiers, quand on a seulement besoin d'exporter un fichier.

Afin d'éviter de se mélanger les pinceaux et de limiter la contagion, nous recommandons de :

- créer **un** dossier d'exportation par projet ;
- nommer ce dossier de façon aussi explicite que possible ; par exemple : *Dossier où Windows peut écrire* ;
- ne jamais partager d'autres dossiers que celui-ci avec le Windows *invité*, mis à part le dossier d'importation que le paragraphe précédent préconise.

page 166 Les recettes « récupérer des fichiers depuis un système virtualisé » et « chiffrer une clé USB » expliquent comment procéder pratiquement.
page 137

Quand le projet est terminé

Quand ce projet est terminé, il faut faire le ménage, mais avant toute chose :

1. l'œuvre résultante est exportée sur le support approprié (papier, VHS, *etc.*), en s'aidant du paragraphe précédent, qui explique comment faire sortir des fichiers du Windows *invité* ;
2. les fichiers de travail sont, si nécessaire, archivés (le cas d'usage suivant traitant, quelle coïncidence, de la question).
page 77

Puis vient l'heure du grand ménage, qui éliminera du système *hôte* le plus possible de traces du projet achevé :

- l'image de disque sale est retirée de VirtualBox et effacée « pour de vrai » ;
- le dossier d'importation est effacé « pour de vrai » ;
- le dossier d'exportation est effacé « pour de vrai »… après avoir vérifié, une dernière fois, que tout ce qui doit être conservé a bien été archivé ailleurs.

page 161 Les recettes « effacer des images de disque » et « effacer des fichiers » expliquent comment effectuer ces opérations.
page 129

Encore un nouveau projet ?

Si un nouveau projet survient, nécessitant lui aussi d'utiliser Windows, ne réutilisons **pas** le même Windows *sale*. Retournons plutôt à l'étape « nouveau projet, nouveau départ ».

page 71

9.6.3 Troisième étape : attaques possibles et contre-mesures

L'hypothèse que nous venons de décrire est basée sur l'utilisation, comme système *hôte*, de la Debian chiffrée mise en place à la première étape du cas d'usage « un nouveau départ ». Toutes les attaques concernant cette Debian chiffrée sont donc applicables à la présente solution. Il est donc maintenant temps d'étudier les attaques praticables contre ce système.

page 58

Traces laissées sur notre Debian chiffrée

La plupart des traces **les plus évidentes** de ce projet sont séparées du reste du système : tous les fichiers de travail sont stockées dans le fichier contenant l'image de disque virtuel. Le nom de la machine virtuelle, sa configuration ainsi que ses périodes d'utilisation laisseront par contre d'autres traces sur notre système Debian.

Si la catastrophe arrive pendant la réalisation du projet Le disque dur de l'ordinateur utilisé contient les fichiers de travail à l'intérieur de l'image de disque virtuel.

Si la catastrophe arrive plus tard L'image de disque virtuel étant convenablement nettoyée lorsque le projet est achevé, **si** la catastrophe (céder face à la loi, découverte d'un problème dans le système cryptographique) arrive **après coup**, les traces résiduelles sur le disque dur seront moins évidentes, et moins nombreuses, que si l'on avait procédé de façon ordinaire.

Même si la catastrophe arrive après la fin du projet, c'est-à-dire : après le nettoyage conseillé ici, il serait malvenu de se sentir immunisé, car comme le début de ce cas d'usage l'explique, l'inconvénient majeur de la méthode décrite ici est qu'elle est basée sur le principe de liste noire, principe abondamment décrié en ces pages… et il restera donc toujours des traces indésirables, auxquelles on n'avait pas pensé, sur le disque dur de l'ordinateur utilisé, en plus de celles qu'on connait bien désormais : journaux, mémoires vive et « virtuelle », sauvegardes automatiques.

page 65
page 53
page 19

Si, malgré ces soucis, l'hypothèse que nous venons de décrire semble être un compromis acceptable, il est maintenant nécessaire de se renseigner sur les limites partagées par toutes les solutions envisagées dans ce cas d'usage.

page 75

Sinon, creusons un peu.

Aller plus loin

Admettons qu'une des attaques décrites à partir de la troisième étape du cas d'usage « un nouveau départ » semble crédible. Si elle réussissait, le contenu du disque dur chiffré du système *hôte* serait lisible, en clair, par l'attaquant. Or nos fichiers de travail sont, rappelons-le, contenus dans l'image de disque virtuel utilisée par notre Windows *invité*… qui est un bête fichier stocké sur le disque dur du système *hôte*. Ces fichiers de travail, ainsi que toute trace enregistrée par les logiciels utilisés dans Windows, deviennent alors lisibles par l'attaquant.

page 59

Nous allons envisager deux pistes permettant de limiter les dégâts. L'une est de type « liste noire », l'autre est de type « liste blanche ».

Stocker l'image de disque virtuel en dehors du disque du système *hôte* Une idée est de stocker hors du disque dur du système *hôte* l'image de disque virtuel utilisée par le système Windows *invité*. Par exemple, sur un disque dur externe chiffré. Ainsi, même si le disque du système *hôte* est déchiffré, nos fichiers de travail restent inaccessibles… pourvu que le disque dur externe qui les contient soit, à ce moment-là, convenablement « rangé ».

page 53 Cette approche est de type « liste noire », avec tous les problèmes que ça pose. Les fichiers de travail et le système Windows sont certes extraits du disque dur du système *hôte*, mais il ne faut pas oublier une chose : ces données seront utilisées par un logiciel page 19 animé par le système *hôte*, nommément : VirtualBox. Comme le chapitre « traces à tous les étages » l'explique, diverses traces subsisteront donc, inévitablement, sur le disque dur **interne** de l'ordinateur utilisé.

Pour suivre cette piste :

page suivante
- se renseigner sur les limites partagées par toutes les solutions envisagées dans ce cas d'usage ;
page 137
- se reporter à la recette permettant de chiffrer un disque dur externe.

Utiliser un système *live* comme système *hôte* Le pendant de cette approche « liste noire » est une solution de type « liste blanche », conjuguant l'utilisation d'un système *live*, et le stockage de l'image de disque virtuel sur un disque dur externe chiffré.

Pour suivre cette piste :

page suivante
- se renseigner sur les limites partagées par toutes les solutions envisagées dans ce cas d'usage ;
page 137
- se reporter à la recette permettant de chiffrer un disque dur externe, et à celle qui explique comment utiliser un système *live*.

page 99

9.7 Nettoyer les métadonnées du document terminé

Une fois notre document terminé, on l'exportera dans un format adapté à l'échange de documents – par exemple un PDF pour imprimer un texte, un fichier AVI ou Ogg pour publier une vidéo sur Internet, *etc.*

Considérons qu'on publie notre document sans prendre de plus amples précautions : un adversaire à qui il déplairait va probablement tout simplement commencer par télécharger le document en quête d'éventuelles métadonnées qui le rapprocheraient de ses auteurs.

Malgré les précautions qu'on a déjà prises, il est bon de nettoyer les éventuelles métadonnées présentes. page 175

9.8 Limites communes à ces politiques de sécurité

Toute politique de sécurité étudiée dans ce cas d'usage est vulnérable à un certain nombre d'attaques. Ce, qu'elle soit basée sur un système *live* ou sur l'envoûtement de l'infâme Windows.

Les étapes 4 et 5 du nouveau départ étudient certaines des attaques imaginables, relevant plus ou moins de la science-fiction, selon l'époque, le lieu, les protagonistes et les circonstances. Le moment est venu de les relire d'un œil nouveau. page 57

Par ailleurs, la partie « problématiques » de ce tome abordait, de façon relativement générale, de nombreux modes de surveillance, qu'il peut être bon de réétudier à la lumière de la situation concrète qui nous occupe ; nommons en particulier les questions d'électricité, champs magnétiques et ondes radios, ainsi que les effets des divers mouchards. page 7 page 14 page 23

Chapitre 10

Cas d'usage : archiver un projet achevé

10.1 Contexte

Un projet sensible touche à sa fin ; par exemple, un livre a été maquetté et imprimé, un film a été monté, compressé, et gravé sur DVD.

page 65

En général, il ne sera dès lors plus nécessaire de pouvoir accéder en permanence aux fichiers de travail (iconographie en haute résolution, *rushes* non compressés). Par contre, il peut être utile de pouvoir les retrouver plus tard, par exemple pour une réédition, une version mise à jour…

Vu qu'un système est d'autant plus susceptible d'être *attaqué* qu'il est fréquemment utilisé, autant extraire les informations rarement utilisées de l'ordinateur utilisé quotidiennement. De surcroît, il est plus facile de nier tout lien avec des fichiers, lorsqu'ils sont stockées sur une clé USB au fond d'un bois, que lorsqu'ils sont rangés sur le disque dur de l'ordinateur familial.

10.2 Est-ce bien nécessaire ?

La première question à se poser avant d'archiver de tels fichiers est la suivante : est-il *vraiment* nécessaire de les conserver ? Lorsqu'on ne dispose plus *du tout* d'une information, quiconque aura beau insister, personne ne sera en mesure de la donner, et c'est parfois la meilleure solution.

10.3 Évaluer les risques

10.3.1 Que veut-on protéger ?

Que donnent les catégories définies lorsque nous parlions d'évaluation des risques, appliquées à ce cas ?

page 49

- confidentialité : éviter qu'un œil indésirable ne tombe trop aisément sur les informations archivées ;
- intégrité : éviter que ces informations ne soient modifiées à notre insu ;
- accessibilité : faire en sorte que ces informations restent accessibles quand on en a besoin.

Ici, l'accessibilité est secondaire par rapport à la confidentialité : toute l'idée de l'archivage est de faire un compromis, en rendant l'accès aux données plus difficile *pour tout le monde*, afin de leur offrir une meilleur confidentialité.

10.3.2 Contre qui veut-on se protéger ?

Les risques envisagés dans notre « nouveau départ » sont valables ici aussi : un

page 57

cambriolage, une perquisition ayant des motifs qui ne sont pas directement liés aux informations qu'on veut ici protéger.

Ajoutons, à ces risques, la possibilité que le livre ou le film produit déplaise à quelque commissaire, ministre, P.D.G. ou assimilé. Ça arrive. Admettons que :

- cette autorité a eu vent d'indices lui permettant de soupçonner qui a commis le chef d'œuvre ;
- cette autorité est en mesure de mandater une cohorte de pénibles hommes en armes et uniforme, au petit matin et au domicile des personnes soupçonnées.

Une telle inopportune intrusion débouchera au minimum, de façon tout aussi fâcheuse qu'évidente, sur la saisie de tout matériel informatique qui pourra y être découvert. Ce matériel sera ensuite remis, par les intrus, à un autre homme de main des autorités, qui pratiquera un genre d'autopsie visant à mettre au jour les données stockées sur ce matériel… ou l'ayant été.

page 32

10.4 Méthode

La méthode la plus simple à l'heure actuelle est :

page 137

1. créer une clé USB ou un disque dur externe chiffré ;
2. copier les fichiers à archiver vers ce périphérique ;

page 127

3. supprimer et écraser le contenu des fichiers de travail.

Une fois ces opérations effectuées, la clé ou le disque dur pourra être entreposé dans un autre lieu que l'ordinateur utilisé couramment.

On pourrait envisager l'utilisation de CD ou de DVD, pour leur faible coût, mais à l'heure actuelle, il est plus complexe de chiffrer correctement des données sur ces supports que sur des clés USB, qui sont désormais monnaie courante et faciles à se procurer.

10.5 Quelle phrase de passe ?

page 91

Vu que les fichiers seront archivés sous forme chiffrée, il sera nécessaire de choisir une phrase de passe. Or, vu que la vocation est l'archivage, cette phrase de passe ne sera pas souvent utilisée. Et une phrase de passe rarement utilisée a toutes les chances d'être oubliée… rendant impossible l'accès aux données.

Face à ce problème, on peut envisager quelques pistes.

10.5.1 Écrire la phrase de passe quelque part

Toute la difficulté étant de savoir où l'écrire, ranger ce document pour pouvoir le retrouver… sans pour autant que d'autres puissent le retrouver et l'identifier comme une phrase de passe.

10.5.2 Utiliser la même phrase de passe que pour son système quotidien

page 105

La phrase de passe de son système quotidien, dans le cas où il est chiffré, est une phrase qu'on tape régulièrement, et dont on a toutes les chances de se souvenir.

Par contre :

- si on est forcé de révèler la phrase de passe commune, l'accès à l'archive devient également possible ;

- il est nécessaire d'avoir **très fortement** confiance dans les ordinateurs avec lequel on accèdera aux archives. Sinon, on peut se faire « piquer », à son insu, la phrase de passe, qui pourra ensuite être utilisée pour lire non seulement les informations archivées, mais aussi toutes les données stockées sur l'ordinateur.

10.5.3 Partager le secret à plusieurs

Il est possible de partager un secret à plusieurs. Cela impose de réunir plusieurs personnes afin de pouvoir accéder au contenu archivé. C'est à peser : ça peut compliquer la tâche aussi bien pour des accès désirés qu'indésirables.

page 149

10.6 Un disque dur ? Une clé ? Plusieurs clés ?

Selon les choix faits précédemment, entre autres sur la phrase de passe, on peut se demander quels supports utiliser. Sachant que sur le plan technique, le plus simple actuellement est d'avoir une seule phrase de passe par support.

Un disque dur externe peut contenir plus de données qu'une clé USB, et est donc parfois nécessaire : pour archiver un projet de vidéo, par exemple.

Archiver plusieurs projets sur un même support permet de se simplifier la tâche, mais il devient alors difficile de séparer les projets selon les niveaux de confidentialité souhaités. Qui plus est, en procédant ainsi, les personnes pouvant accéder aux archives d'un projet ont aussi accès aux autres, ce qui n'est pas forcément souhaitable.

Par ailleurs, si la phrase de passe est un secret partagé, autant faciliter l'accès aux personnes partageant le secret, en ayant un support qu'elles peuvent se transmettre.

Troisième partie

Outils

Dans cette troisième partie, nous expliquerons comment appliquer concrètement quelques-unes des pistes évoquées précédemment.

Cette partie n'est qu'une annexe technique aux précédentes : une fois comprises les problématiques liées à l'intimité dans le monde numérique ; une fois les réponses adaptées choisies, reste la question du « Comment faire ? », à laquelle cette annexe apporte certaines réponses.

Du bon usage des recettes

Les outils et recettes qui suivent sont des solutions extrêmement partielles, qui ne sont d'aucune utilité tant qu'elles ne font pas partie d'un ensemble de pratiques articulées de façon cohérente.

Piocher dans cette boîte à outils sans avoir, au préalable, étudié la partie sur le choix d'une réponse adaptée et défini une *politique de sécurité*, est un moyen remarquable de se tirer une balle dans le pied en croyant, à tort, avoir résolu tel ou tel problème.

page 47

On ne peut pas faire plaisir à tout le monde

Partons du principe, pour la plupart des recettes présentées dans ce guide, que l'on utilise GNU/Linux avec le bureau GNOME ; elles ont été écrites et testées sous Debian GNU/Linux version 6.0 (surnommée Squeeze) [1] et Tails [2] (*The Amnesic Incognito Live System*).

Pour autant, ces recettes sont généralement concoctables avec d'autres distributions basées sur Debian, telles qu'Ubuntu [3] ou gNewSense [4].

Si l'on n'utilise pas encore GNU/Linux, ou pourra consulter le cas d'usage un nouveau départ ou utiliser un système *live*.

page 57
page 99

1. http://www.debian.org/releases/squeeze/
2. https://tails.boum.org/
3. http://www.ubuntu-fr.org/
4. http://www.gnewsense.org/Main.fr/HomePage

De la bonne interprétation des recettes

Avant de passer aux recettes elles-mêmes, quelques remarques transversales nous ont paru nécessaires.

Dans un certain nombre d'outils, les procédures sont présentées pas à pas, et expliquent, chaque fois que c'est possible, le sens des actions que l'on propose d'effectuer. Une utilisation efficace de ces outils nécessite de s'entendre sur quelques points :

- L'ordre dans lequel chaque recette est développée est d'une importance capitale. Sauf mention contraire, il est simplement inimaginable de sauter une étape pour ensuite revenir en arrière : le résultat, si jamais ces opérations désordonnées en donnaient un, pourrait être soit différent de celui escompté, soit tout bonnement catastrophique.
- Dans le même ordre d'idée, les actions indiquées doivent être effectuées à la lettre. Omettre une option, ouvrir le mauvais dossier, peut avoir pour effet de totalement modifier le sens ou les effets d'une recette.
- De manière générale, la bonne compréhension de ces recettes demande d'y accorder un minimum d'attention et de vivacité d'esprit. On ne peut pas tout réexpliquer à chaque fois : il est implicite d'avoir auparavant « suivi » et intégré les explications des « cas d'usage » dont ces recettes ne sont que la dernière étape.

Enfin l'exemplaire que vous avez entre les mains n'est pas forcément à jour aux vues des versions actuelles des différents outils impliqués. La version en ligne [https://guide.boum.org/] du Guide d'Autodéfense Numérique a des chances d'être plus à jour.

Chapitre 11

Utiliser un terminal

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 15 à 30 minutes.

Souvent, on utilise un ordinateur personnel en cliquant sur des menus et des icônes. Cependant, il existe une autre façon de lui « parler » : en tapant des bouts de texte que l'on appelle des « commandes ». On appelle cette façon d'interagir avec un ordinateur « le terminal », « le *shell* » ou encore « la ligne de commande ».

Ce guide cherche le plus souvent possible à contourner l'utilisation de cet outil, qui est assez déroutant lorsque l'on n'y est pas habitué. Cependant, son usage s'est parfois avéré indispensable.

11.1 Qu'est-ce qu'un terminal ?

Une explication détaillée sur l'usage de lignes de commandes n'est pas l'objet de ce guide, et Internet regorge de tutoriels et de cours assurant très bien ce rôle [1]. Il semblait cependant nécessaire de poser quelques bases sur la manière de s'en servir.

Alors on va tout simplement commencer par ouvrir un terminal : sur un bureau GNOME standard, il suffit de cliquer sur *Applications* → *Accessoires* → *Terminal*. Apparaît alors une fenêtre qui indique :

```
IDENTIFIANT@LE_NOM_DE_LA_MACHINE :~$
```

À la fin se trouve un carré, appelé « curseur », qui correspond à l'endroit où inscrire le texte de la commande. Concrètement, avec l'identifiant *roger* sur une machine nommée *debian*, on aura sous les yeux :

```
roger@debian :~$ 
```

C'est à partir de cet état, appelé « invite de commande », que l'on peut taper directement les commandes qu'on veut faire exécuter à l'ordinateur.

L'effet final de ces commandes est souvent le même que celui qu'on peut obtenir en cliquant au bon endroit dans une interface graphique.

1. Entre autres, une page sur *ubuntu-fr.org* [http://doc.ubuntu-fr.org/console] qui se termine elle-même par d'autres liens.

Par exemple, si dans le terminal qu'on vient d'ouvrir, on écrit juste `gedit` puis qu'on tape sur *Entrée*, le résultat est qu'on ouvre un éditeur de texte. On aurait pu faire exactement la même chose en cliquant sur *Applications* → *Accessoires* → *Éditeur de texte gedit*. Par contre, on ne pourra pas entrer de nouvelle commande dans notre terminal tant que l'on aura pas quitté l'éditeur de texte.

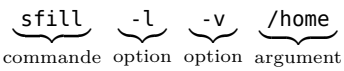
Dans le cadre de ce guide, l'intérêt du terminal est surtout qu'il permet d'effectuer des actions qu'aucune interface graphique ne propose pour le moment.

11.2 À propos des commandes

Les commandes sont comme des ordres qu'on donne à l'ordinateur par le biais du terminal. Ces « lignes de commande » ont leur propre langage, avec leurs mots, leurs lettres, et leur syntaxe. Quelques remarques à ce sujet sont donc utiles.

11.2.1 Syntaxe

Prenons par exemple cette commande, `sfill`, qui permet à peu près les mêmes opérations que `nautilus-wipe`, un outil graphique qui sera présenté plus tard page 135 :



Dans cette ligne de commande, on peut voir, dans l'ordre :

- la *commande* que l'on appelle est `sfill`. La commande est en général un programme installé sur le système ;
- deux *options*, `-l` et `-v` qui modifient le comportement du programme `sfill`. Ces dernières peuvent être facultatives selon le programme (et commencent par un ou deux tiret pour qu'on les distingue) ;
- un *argument* `/home` qui précise ce sur quoi va travailler la commande. Il peut y en avoir plusieurs, ou aucun, tout dépend de la commande.

Chacun de ces éléments doit être séparé des autres par un (ou plusieurs) espace(s). Il y a donc un espace entre la commande et la première option, entre la première option et la suivante, entre la dernière option et le premier argument, entre le premier argument et les suivants, *etc.*

Pour savoir quelles sont les commandes disponibles, leurs options et leurs arguments, pas de mystère : chaque commande dispose normalement d'une page de manuel. Pour y accéder, il suffit d'aller dans *Système* → *Aide*, puis dans *Pages de manuel*. Ces dernières peuvent toutefois être difficiles à comprendre par leur aspect technique, et ne sont parfois disponibles qu'en anglais.

11.2.2 Insertion du chemin d'un fichier

Lors de l'utilisation d'un terminal, on a souvent besoin d'indiquer des dossiers et des fichiers. On parle de « chemin » car on décrit généralement dans quel dossier et sous-dossier un fichier se trouve. Pour séparer un dossier de ce qu'il contient, on utilise le caractère `/` (qui se prononce « slash »).

Pour donner un exemple, voici le *chemin* du document `recette.txt` qui se trouve dans le dossier `Documents` du dossier personnel du compte `alligator` :

```
/home/alligator/Documents/recette.txt
```

Comme beaucoup de commandes attendent des noms de fichiers comme arguments, cela devient vite fastidieux de taper leurs chemins complets à la main. Il y a cependant un moyen simple d'insérer un chemin : quand on attrape avec la souris l'icône d'un fichier, et qu'on le déplace pour le lâcher sur le terminal, son chemin s'écrit là où se trouve le curseur.

Cela ne marche cependant qu'avec les vrais fichiers ou dossiers. On obtiendra un nom bizarre qui ne fonctionnera pas, par exemple, pour les fichiers mis à la corbeille, l'icône du *Dossier personnel* sur le bureau ou avec les icônes de clés USB.

11.2.3 Exécution

Une fois que l'on a tapé une commande, on demande à l'ordinateur de l'« exécuter » en appuyant sur la touche *Entrée*.

11.2.4 Fin ou interruption de la commande

L'exécution de la commande prend plus ou moins de temps. Lorsqu'elle est terminée, le terminal retourne toujours à l'état où il était avant qu'on lance la commande, l'« invite de commande » :

```
roger@debian :~$
```

On dit alors que le terminal « rend la main ».

Si on souhaite interrompre l'exécution d'une commande avant qu'elle soit terminée, on peut appuyer la touche Ctrl, et tout en laissant cette touche enfoncée appuyer sur la touche C. On arrête alors la commande immédiatement, un peu comme quand on ferme la fenêtre d'un programme.

11.2.5 Typographie

La plupart des symboles utilisés pour entrer les commandes complètes sont des symboles courants. Lorsqu'une commande emploie le symbole « - », il ne s'agit que du « tiret » qu'on peut obtenir en tapant (sur un clavier français) la touche 6. Pour un « ' » (apostrophe droite), c'est le 4 …

D'autres symboles sont rarement utilisés en dehors du terminal, mais sont disponibles avec les claviers standards. Ils sont mêmes indiqués sur le clavier, et accessibles à l'aide de la touche Alt de droite, notée AltGr. Voici, en se basant sur un clavier de PC français standard, la correspondance de quelques touches avec les symboles qu'elles écrivent, et leur nom (bien peu seront en fait utilisées dans ce guide) :

Touches	Résultat	Nom du symbole
AltGr + 2	~	tilde
AltGr + 3	#	dièse
AltGr + 4	{	accolade gauche
AltGr + 5	[	crochet gauche
AltGr + 6	|	*pipe*
AltGr + 8	\	antislash
AltGr + 0	@	arobase
AltGr +)	]	crochet droit

Touches	Résultat	Nom du symbole
AltGr + =	}	accolade droite

11.2.6 Noms à remplacer

Parfois, on précise que l'on va nommer quelque chose que l'on a trouvé pour le réutiliser plus tard. Par exemple, on dira que l'identifiant est `LOGIN`. Mettons qu'on travaille sous l'identifiant `paquerette`. Lorsqu'on écrira « taper `LOGIN` en remplaçant `LOGIN` par l'identifiant de son compte », il faudra taper en réalité `paquerette`. Si l'on tape `LOGIN`, cela ne fonctionnera pas…

11.3 Terminal ? Terminal administrateur ?

Dans le menu *Applications* → *Accessoires* se trouvent deux entrées permettant d'obtenir un terminal : *Terminal* et *Terminal administrateur*.

La première permet d'obtenir un terminal fonctionnant avec les droits d'accès de la session en cours. On ne pourra donc pas l'utiliser pour effectuer des opérations *privilégiées* comme créer une partition chiffrée. Le symbole à la fin de l'« invite de commande » sera un dollar (`$`).

La seconde commande permet d'obtenir un terminal avec les droits d'administration. On appelle également cela un *shell root*. À partir de ce dernier, les commandes pourront accéder à l'intégralité du système, sans restriction… avec les risques que cela comporte, donc. Le symbole à la fin de l'« invite de commande » sera un dièse (`#`).

11.4 Encore une mise en garde

Plus encore que pour les recettes dont on parlait plus haut, les commandes doivent être tapées très précisément. Oublier un espace, omettre une option, se tromper de symbole, être imprécis dans un argument, c'est changer le sens de la commande.

Et comme l'ordinateur effectue *exactement* ce qui est demandé, si on change la commande, il fera *exactement autre chose*…

11.5 Un exercice

On va créer un fichier vide nommé « essai », qu'on va ensuite supprimer (sans recouvrir son contenu).

Dans un terminal, entrer la commande :

```
touch essai
```

Et taper sur *Entrée* pour que l'ordinateur l'exécute.

La *commande* `touch` donne l'ordre de créer un fichier vide ; l'*argument* `essai` donne le nom de ce fichier. Aucune option n'est utilisée.

On peut alors vérifier que ce fichier a été créé en lançant la commande `ls` (qui signifie « lister ») :

```
ls
```

Une fois la commande lancée, l'ordinateur répond avec une liste. Sur celui utilisé pour les tests, cela donne :

```
Bureau
essai
```

`Bureau` est le nom d'un dossier qui existait déjà avant, et `essai` le nom du fichier qu'on vient de créer. Un autre ordinateur auraient pu répondre avec de nombreux autres fichiers en plus de `Bureau` et de `essai`.

Ce que répond la commande `ls` n'est qu'une autre manière de voir ce que l'on peut obtenir par ailleurs. En cliquant, sur le bureau, sur l'icône du *Dossier personnel*, on pourra noter dans le navigateur de fichiers l'apparition d'une nouvelle icône représentant le fichier `essai` que l'on vient juste de créer…

On va maintenant supprimer ce fichier. La ligne de commande pour le faire a pour syntaxe générale :

```
rm [options] NOM_DU_FICHIER_A_SUPPRIMER
```

On va utiliser l'option `-v` qui, dans le cadre de *cette* commande, demande à l'ordinateur d'être « bavard » (on parle de « mode verbeux ») sur les actions qu'il va effectuer.

Pour insérer le nom du fichier à supprimer, on va utiliser l'astuce donnée précédemment pour indiquer le chemin du ficher. On va donc :

- taper `rm -v` dans notre terminal,
- taper un espace afin de séparer l'option `-v` de la suite,
- dans la fenêtre du *Dossier personnel*, on va prendre avec la souris l'icône du fichier `essai` et la déposer dans le terminal.

À la fin de cette opération, on doit obtenir quelque chose comme :

```
rm -v '/home/LOGIN/essai'
```

On peut alors appuyer sur la touche *Entrée* et constater que l'ordinateur répond :

```
« /home/LOGIN/essai » supprimé
```

Cela indique qu'il a bien supprimé le fichier demandé. On peut encore vérifier son absence en lançant un nouveau `ls` :

```
ls
```

On doit constater l'absence de `essai` dans la liste que nous répond la commande. Sur le même ordinateur que tout à l'heure, cela donne :

```
Bureau
```

Et l'icône doit également avoir disparu dans le navigateur de fichiers. Apparemment, il a été supprimé… même si, comme expliqué dans la première partie, son contenu existe encore sur le disque. Comme c'était un fichier vide nommé « essai », on peut se dire que ce n'est pas bien grave.

page 32

11.6 Attention aux traces !

page 21

La plupart des *shells* enregistrent automatiquement les lignes de commande que l'on a tapées dans un fichier « d'historique ». C'est bien pratique pour retrouver plus tard des commandes que l'on a pu utiliser, mais cela laisse également sur le disque une trace de nos activités.

Le *shell* standard dans Debian s'appelle `bash`. Avec ce dernier, pour désactiver temporairement l'enregistrement de l'historique dans le terminal que l'on utilise, il suffit de faire :

```
unset HISTFILE
```

page 129

Par ailleurs, les commandes sont enregistrées dans le fichier caché `.bash_history` (qui se trouve dans le *Dossier personnel*). On peut donc avoir envie de le nettoyer de temps en temps.

11.7 Pour aller plus loin

Cette première expérience avec cette fenêtre pleine de petits caractères pourrait être le début d'une longue passion. Pour l'entretenir, rien de mieux que de prendre le temps de lire le chapitre « Débuter en console[2] » de la *formation Debian* ou celui baptisé « Linux en mode texte : consolez-vous ![3] » du livre *Linux aux petits oignons*.

2. http://formation-debian.via.ecp.fr/debuter-console.html
3. http://www.editions-eyrolles.com/Chapitres/9782212124248/Pages-63-82_Novak.pdf

Chapitre 12

Choisir une phrase de passe

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 10 minutes environ.

Une « phrase de passe » (ou *passphrase* en anglais) est un secret qui sert à protéger des données chiffrées. C'est ce qu'on utilise pour chiffrer un disque dur, des emails, des documents… voire comme nous le verrons dans le second tome de cet ouvrage, des clés cryptographiques.

On parle de *phrase* plutôt que de *mot* de passe car un seul *mot*, aussi bizarre et compliqué soit-il, est beaucoup moins résistant qu'une simple phrase de plusieurs mots. On considère qu'une phrase de passe doit être constituée d'au moins 10 mots. Mais plus il y en a, mieux c'est !

Un critère important, mais parfois négligé : une bonne phrase de passe est une phrase de passe dont on peut *se souvenir*[1], ça évite de la noter sur un papier, grave erreur qui rend caduc l'intérêt de se faire une phrase de passe béton. Mais, et c'est tout aussi important, une bonne phrase de passe doit être *impossible à deviner*. Évitons donc les phrases de passe formées de 15 mots composés de caractères aléatoires qu'on aura oublié à peine 15 minutes après l'avoir trouvé.

Une technique simple pour trouver une bonne phrase de passe, difficile à deviner, mais néanmoins facile à retenir, est de fabriquer une phrase qui n'est pas issue d'un texte existant. En effet, que ce soit des paroles de chansons, le vers d'un poème, ou une citation d'un livre, des outils comme le projet Gutenberg[2] rendent de plus en plus facile le test de phrases de passe tirées de la littérature existante[3].

Même s'il faut ici vous en remettre à votre imagination, nous pouvons quand même donner quelques pistes quant aux bonnes habitudes à avoir lors du choix d'une phrase de passe.

1. Fabriquons tout d'abord une phrase dont on se souviendra aisément. Faisons tourner nos méninges un instant.
2. Trouvons dans cette phrase ce que l'on peut modifier pour la rendre plus difficilement devinable. On peut ainsi penser à y ajouter de l'argot, des mots de

1. Randall Munroe, 2014, *Password Strength* [https://xkcd.com/936/] (en anglais).
2. Wikipédia, 2014, *Projet Gutenberg* [https://fr.wikipedia.org/wiki/Projet_Gutenberg].
3. Dan Goodin, 2013, *How the Bible and YouTube are fueling the next frontier of password cracking* [http://arstechnica.com/security/2013/10/how-the-bible-and-youtube-are-fueling-the-next-frontier-of-password-cracking/] (en anglais).

différentes langues, mettre des majuscules là où l'on ne les attend pas remplacer des caractères par d'autres, laisser libre cours à son imagination quant à l'orthographe *etc.*

Un conseil toutefois : il est préférable d'éviter les caractères accentués ou tout autre symbole n'étant pas directement disponible sur un clavier américain. Cela peut éviter des problèmes de touches absentes ou difficiles à retrouver, et surtout de mauvais codage des caractères, si l'on est amené à taper notre phrase de passe sur un clavier différent de celui dont on a l'habitude.

Un exemple, prenons cette phrase sans sens apparent :

correct cheval pile agraphe

On peut les transformer ainsi, pour obtenir une meilleure phrase de passe :

korect sheVall-peEla ! grafF

Une fois vos données chiffrées avec votre nouvelle phrase de passe, c'est une bonne idée de l'utiliser tout de suite une bonne dizaine de fois pour déchiffrer vos données. Cela permettra d'apprendre un peu à vos doigts comment la taper et ainsi de la mémoriser à la fois mentalement et physiquement.

N'oublions toutefois pas que si trouver une telle phrase de passe n'est pas sans effort, cela ne dispense aucunement d'en trouver une différente par support que l'on chiffre. L'usage d'une même phrase de passe, ou pire d'un même mot de passe, pour une variété de choses différentes, boîtes mail, compte *PayPal*, banque en ligne *etc.* peut rapidement s'avérer désastreux si elle est dévoilée.

Chapitre 13

Démarrer sur un CD, un DVD ou une clé USB

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 1 minute à 20 minutes environ.

On va voir ici comment démarrer un ordinateur PC sur un média externe, par exemple un CD d'installation de Debian, ou un système *live* sur une clé USB ou même une carte SD.

Parfois, en particulier sur les ordinateurs modernes, c'est très simple. D'autres fois, c'est un peu à s'arracher les cheveux…

Cela se joue au tout début du démarrage de l'ordinateur, dans le micrologiciel. On a vu que c'est lui qui permet de choisir le périphérique (disque dur, clé USB, CD-ROM ou DVD, *etc.*) où se trouve le système qu'on veut utiliser. page 13

13.1 Essayer naïvement

Commencer par mettre le CD ou le DVD dans le lecteur, ou par brancher la clé, puis (re)démarrer l'ordinateur. Parfois, ça marche tout seul. Si c'est le cas, c'est gagné, lire la suite est inutile !

13.2 Tenter de choisir le périphérique de démarrage

Sur les micrologiciels récents, il est souvent possible de choisir un périphérique de démarrage au cas par cas.

(Re)démarrer l'ordinateur en regardant attentivement les tout premiers messages qui s'affichent à l'écran. Chercher des messages en anglais qui ressembleraient à :

- `Press [KEY] to select temporary boot device`
- `[KEY] = Boot menu`
- `[KEY] to enter MultiBoot Selection Menu`

Ces messages disent d'utiliser la touche `KEY` pour choisir un périphérique de démarrage. Cette touche est souvent `F12` ou `F10`.

Sur les Mac, il existe un équivalent de cette possibilité : immédiatement après l'allumage de l'ordinateur, il faut appuyer et maintenir la touche alt (parfois également

marquée option). Au bout d'un moment, on doit normalement voir apparaître le *Gestionnaire de démarrage*[1].

Mais revenons à nos PC. Souvent, le micrologiciel va trop vite, on n'a pas le temps de lire le message, de le comprendre et d'appuyer sur la touche. Qu'à cela ne tienne, une fois la bonne touche identifiée, redémarrer encore la machine et appuyer sur la touche en question (ne pas maintenir la touche enfoncée, mais la presser puis la relâcher plusieurs fois) dès l'allumage de l'ordinateur.

Avec un peu de chance, un message comme celui-ci s'affiche :

```
+---------------------------------+
| Boot Menu                       |
+---------------------------------+
|                                 |
| 1: USB HDD                      |
| 4: IDE HDD0: BDS GH87766319819  |
| 8: Legacy Floppy Drives         |
|                                 |
|    <Enter Setup>                |
|                                 |
+---------------------------------+
```

Si ça marche, c'est gagné. Choisir la bonne entrée dans ce menu, en se déplacant avec les flèches du clavier ↑ et ↓, puis appuyer sur *Entrée*. Par exemple, pour démarrer sur une clé USB, choisir `USB HDD`. L'ordinateur doit démarrer sur le périphérique sélectionné. Lire la suite est inutile !

13.3 Modifier les paramètres du micrologiciel

Si choisir un périphérique de démarrage temporaire ne fonctionne pas, il va falloir rentrer dans le micrologiciel pour choisir manuellement l'ordre de démarrage. Pour pimenter un peu la chose, les micrologiciels sont quasiment tous différents, de telle sorte qu'il est impossible de donner une recette qui marche systématiquement[2].

13.3.1 Entrer dans l'intreface de configuration du micrologiciel

Encore une fois, il s'agit de (re)démarrer l'ordinateur en regardant attentivement les premiers messages qui s'affichent à l'écran. Chercher des messages en anglais qui ressembleraient à :

- `Press [KEY] to enter setup`
- `Setup: [KEY]`
- `[KEY] = Setup`
- `Enter BIOS by pressing [KEY]`
- `Press [KEY] to enter BIOS setup`
- `Press [KEY] to access BIOS`
- `Press [KEY] to access system configuration`
- `For setup hit [KEY]`

Ces messages disent d'utiliser la touche `[KEY]` pour entrer dans le micrologiciel. Cette touche est souvent Suppr (*Delete*, *DEL*) ou F2 , parfois F1 , F10 , F12 , Échap , ⇆ (*Tab*), voire autre chose encore.

1. http://support.apple.com/kb/HT1310?viewlocale=fr_FR
2. Des tutoriels illustrés pour quelques BIOS sont disponibles sur cette page [http://www.hiren.info/pages/bios-boot-cdrom]

Voici un tableau qui résume les touches d'accès au micrologiciel pour quelques fabriquants d'ordinateurs communs[3].

Fabriquant	**Modèle**	**Touches observées**
Acer	modèles récents	F2 , Suppr
Acer	modèles anciens	Ctrl + Alt + Échap , F1
AST, ARI		Ctrl + Alt + Échap , Ctrl + Alt + Suppr
Compaq	modèles récents	F10
Compaq	modèles anciens	F1 , F2 , Suppr
CompUSA		Suppr
Cybermax		Échap
Dell	modèles récents	F2
Dell	anciens *desktops*	Ctrl + Alt + Entrée , Suppr
Dell	anciens portables	Fn + Échap , Fn + F1
eMachines		⇤⇥ (*Tab*), Suppr , F2
Fujitsu		F2
Gateway		F1 , F2
HP		F1 , F2 , Échap
HP	tablet PC	F10 , F12
IBM	modèles récents	F1
IBM	anciens modèles	F2
IBM/Lenovo	modèles récents	F1 , F2
IBM/Lenovo	anciens modèles	Ctrl + Alt + F3 , Ctrl + Alt + Ins , Fn + F1
Intel	Tangent	Suppr
Micron		F1 , F2 , Suppr
NEC		F2
Packard Bell		F1 , F2 , Suppr
Shuttle		F1 , Suppr
Sony		F1 , F2 , F3
Tiger		Suppr
Toshiba		F1 , Échap
Toshiba	Equium	F12

Souvent, le micrologiciel va trop vite, et on n'a pas le temps de lire le message, de le comprendre et d'appuyer sur la touche. Qu'à cela ne tienne, une fois la bonne touche identifiée, redémarrer encore la machine en appuyant sur la touche en question (ne pas

3. Tim Fisher, 2014, *BIOS Setup Utility Access Keys for Popular Computer Systems* [http://pcsupport.about.com/od/fixtheproblem/a/biosaccess_pc.htm] ainsi que MichaelStevensTech, 2014, *How to access/enter Motherboard BIOS* [http://michaelstevenstech.com/bios_manufacturer.htm] (liens en anglais).

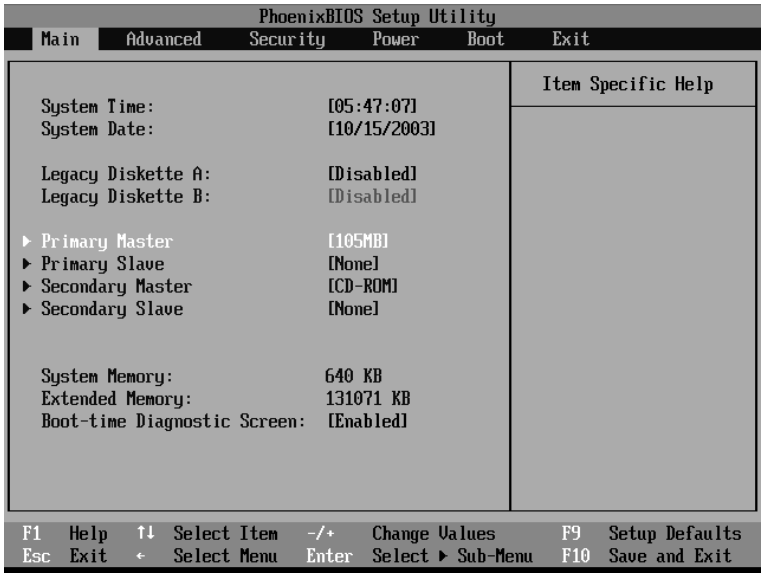
maintenir la touche enfoncée, mais la presser puis la relâcher plusieurs fois). Parfois, l'ordinateur se perd et plante. Dans ce cas, redémarrer et réessayer…

Si une image s'affiche à la place du message que l'on espère voir, il se peut que le micrologiciel soit configuré pour afficher un logo plutôt que ses messages. Essayer d'appuyer sur Échap ou sur ⇆ (*Tab*) pour voir les messages.

Si l'ordinateur démarre trop rapidement pour qu'on ait le temps de lire les messages qu'il affiche, il est parfois possible d'appuyer sur la touche *Pause* (souvent en haut à droite du clavier) pour geler l'écran. Réappuyer sur n'importe quelle touche peut « dégeler » l'écran.

13.3.2 Modifier la séquence de démarrage

Une fois dans le micrologiciel, l'écran est souvent bleu ou noir, et plein de menus. En général, une zone en bas ou à droite de l'écran explique comment naviguer entre les options, comment changer d'onglet… Elle est souvent en anglais : aide se dit « *help* », touche se dit « *key* », sélectionner se dit « *select* », valeur « *value* » et modifier « *modify* ». Les touches à utiliser pour se déplacer sont généralement décrites aussi, par exemple `←↑↓→: Move` (en anglais, déplacer se dit « *move* »). Il s'agit des flèches du clavier ↓ et ↑ et/ou ← et →. Parfois, la touche ⇆ (*Tab*) est utile aussi.


PhoenixBIOS Setup Utility
Main Advanced Security Power Boot Exit
Item Specific Help
System Time: [05:47:07]
System Date: [10/15/2003]
Legacy Diskette A: [Disabled]
Legacy Diskette B: [Disabled]
Primary Master [105MB]
Primary Slave [None]
Secondary Master [CD-ROM]
Secondary Slave [None]
System Memory: 640 KB
Extended Memory: 131071 KB
Boot-time Diagnostic Screen: [Enabled]
F1 Help ↑↓ Select Item -/+ Change Values F9 Setup Defaults
Esc Exit ← Select Menu Enter Select ▸ Sub-Menu F10 Save and Exit

Un écran de BIOS

L'idée, c'est de fouiller dedans jusqu'à trouver quelque chose qui contient `boot`, et qui ressemble par exemple à :

- `First Boot Device`
- `Boot Order`
- `Boot Management`
- `Boot Sequence`

S'il n'y a pas, tenter quelque chose comme `Advanced BIOS Features` (sur les *AwardBIOS*) ou `Advanced features` (sur les *AMIBIOS*).

Une fois la bonne entrée trouvée, il s'agit de trouver comment on la modifie. Par exemple `Enter: Select` ou `+/-: Value`. L'objectif est alors de mettre le CD/DVD ou l'USB en premier, selon sur lequel on veut démarrer.

Parfois, il faut entrer dans un sous-menu. Par exemple s'il y a un menu `Boot order` et qu'il est écrit dans l'aide `Enter: Select`, appuyer sur entrée une fois le menu sélectionné.

D'autres fois, les options se changent directement. Par exemple, s'il y a une option comme `First boot device` et qu'il est écrit dans l'aide `+/-: Value`, appuyer sur la touche `+` ou la touche `-` jusqu'à ce que la bonne valeur, comme par exemple `IDE DVDROM`, soit sélectionnée. Parfois, ce sont plutôt les touches *Page suivante* ou `PgUp` et *Page précédente* ou `PgDown` qui sont utilisées. Parfois encore, ce sont des touches comme `F5` et `F6`. D'autres fois encore, ces touches servent à monter et à descendre le périphérique dans une liste correspondant à l'ordre de démarrage.

13.3.3 Bien choisir sa nouvelle configuration

Une fois qu'on a réussi à sélectionner le bon support pour le démarrage, il faut se demander si on veut le laisser pour toujours ou pas. Si on veut le laisser, il peut être utile de placer le disque dur en deuxième position dans la séquence de démarrage. Ainsi, si le support placé en premier est absent, l'ordinateur démarrera sur le disque dur. Si l'on ne met pas le disque dur dans la séquence de démarrage, l'ordinateur ne démarrera pas dessus, même en l'abscence de CD, de DVD ou de clé USB.

Cependant, le fait de laisser son ordinateur démarrer *a priori* sur un support externe peut avoir des conséquences fâcheuses : il devient un peu plus facile pour un intrus de le faire démarrer en utilisant ce support, par exemple pour effectuer une attaque.

On peut certes mettre en place, avec le micrologiciel, un mot de passe d'accès à l'ordinateur, qui devra être entré avant tout démarrage. Mais il est inutile de compter sur celui-ci pour protéger quoi que ce soit : cette protection peut, la plupart du temps, être contournée très facilement.

13.3.4 Enregistrer et quitter

Une fois la nouvelle configuration établie, il reste à enregistrer et à quitter. Encore une fois, lire l'aide à l'écran, comme `F10: Save`. Parfois, il faut appuyer une ou plusieurs fois sur Échap pour avoir le bon menu. Un message s'affiche alors pour demander (en anglais) si on est sûr de vouloir enregistrer et quitter. Par exemple :

```
+-------------------------------------+
|         Setup Confirmation          |
+-------------------------------------+
|                                     |
| Save configuration and exit now     |
|                                     |
|     <Yes>              <No>         |
|                                     |
+-------------------------------------+
```

On veut effectivement enregistrer, donc on sélectionne `Yes` et on appuye sur *Entrée*.

Chapitre 14

Utiliser un système *live*

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 30 minutes à une heure, plus environ 30 minutes de téléchargement.

Un système *live* est un système GNU/Linux qui fonctionne sans être installé sur le disque dur de l'ordinateur.

Attention, cela ne signifie pas qu'il n'y aura pas de traces sur le disque dur : par exemple, nombre de systèmes *live* utilisent l'espace d'échange (*swap*) présent sur le disque dur s'ils en détectent un. De plus, ils utilisent parfois automatiquement les partitions qu'ils y détectent.

page 17

14.1 Des systèmes *live* discrets

Par contre, certains systèmes *live* sont spécialement conçus pour (tenter de) ne laisser aucune trace sur le disque dur de l'ordinateur sur lequel ils sont utilisés, à moins que l'on ne leur demande expressément de le faire. C'est par exemple le cas de Tails (*The Amnesic Incognito Live System* — le système *live* amnésique incognito).

Il n'y a alors (si les personnes à l'origine du système *live* ne se sont pas trompées) rien d'écrit sur le disque dur. Tout ce qui sera fait à partir du système *live* sera uniquement écrit en mémoire vive, qui s'efface plus ou moins pour de vrai toute seule quand on éteint l'ordinateur, du moins au bout d'un certain temps.

page 11

Utiliser de tels systèmes *live* est donc l'une des meilleures façons d'utiliser un ordinateur sans laisser de traces. Nous verrons ici comment obtenir un système *live*, et comment démarrer dessus.

Le moyen usuel d'utiliser un système *live* est de l'installer sur une clé USB, une carte SD ou de le graver sur un DVD. On parle dans ce cas de *Live DVD*.

Néanmoins, vu qu'il est possible d'écrire des données sur une clé USB ou une carte SD alors que ce n'est pas possible sur un DVD, après installation du système *live* on a moins de garanties au cas où les gens qui ont écrit le système *live* auraient commis des erreurs. Cela devient aussi plus simple pour des personnes malveillantes de modifier votre système *live* pour, par exemple, enregistrer vos mots de passe ou vos frappes au clavier.

14.2 Télécharger et installer Tails

On va expliquer ici comment télécharger la dernière version de Tails à partir de son site web officiel, vérifier son authenticité et l'installer sur une clé USB, une carte SD ou la graver sur un DVD.

page suivante

Si l'on dispose déjà de la dernière version de Tails, il est possible de la dupliquer simplement. Suivre pour cela l'outil cloner Tails.

Pour télécharger et installer Tails, on va suivre la documentation officielle, disponible sur la page web https://tails.boum.org/download/index.fr.html.

Attention : ce guide fournit des explications sur la vérification de l'intégrité de l'image de Tails. Lorsqu'on arrive à la section « Vérifier l'image ISO » de la documentation officielle de Tails, se rapporter à vérifier son authenticité.

plus bas

14.2.1 Télécharger Tails

Tails peut être téléchargé soit directement avec un navigateur web (en HTTP), soit à l'aide de BitTorrent.

BitTorrent est un protocole de partage de fichiers de pair-à-pair (« *peer-to-peer* » en anglais) ; il permet donc à tous les ordinateurs qui effectuent le téléchargement de participer à la diffusion des fichiers. Cela nécessite d'utiliser un logiciel dédié au téléchargement.

Quand cela est possible, cette méthode a l'avantage de garantir une certaine disponibilité des fichiers au cas où un problème arriverait sur les gros serveurs offrant le téléchargement direct. C'est donc celle-ci que nous allons développer.

plus bas

Si l'on choisit de télécharger l'image directement avec son navigateur web, on peut directement aller vérifier son authenticité.

Télécharger le *torrent*

Pour télécharger en pair-à-pair, il faut tout d'abord télécharger un petit fichier, appelé *torrent*. Ce fichier contient les informations qui seront nécessaires au logiciel de téléchargement pour trouver les sources des fichiers que l'on veut obtenir.

On va donc télécharger le fichier Torrent en cliquant sur le bouton correspondant.

Télécharger l'image du système *live*

Sur une Debian standard, le logiciel de téléchargement de *torrents Transmission* démarrera automatiquement après le téléchargement du fichier torrent. Une fenêtre indiquant les fichiers qui seront téléchargés s'ouvre ; après avoir vérifié le dossier de destination, il suffit de cliquer sur *Ouvrir* pour démarrer le téléchargement.

Si le client BitTorrent ne s'ouvre pas tout seul, on va l'ouvrir à la main :

page 120

- sous Debian ou Ubuntu, ouvrir le *Client BitTorrent Transmission*. Il peut être nécessaire, au préalable, d' installer le paquet `transmission-gtk` ;
- sous Mac OS X, il est aussi possible d'installer Transmission [1] ;
- sous Windows, il est possible d'installer le client libre Vuze [2].

14.2.2 Vérifier l'authenticité du système live

tome 2 ch. 18

L'image du système *live* que l'on vient de télécharger est signée numériquement avec OpenPGP.

1. http://www.transmissionbt.com/
2. http://www.vuze.com/

Après avoir téléchargé l'image ISO de la dernière version de Tails, ainsi que sa signature numérique, il va nous falloir vérifier son intégrité.

tome 2 § 6.3

Importer ensuite la clé OpenPGP qui signe les ISO de Tails :

tome 2 § 18.1

```
Tails developers (signing key) <tails@boum.org> 0xBE2CD9C1
```

tome 2 § 18.8

Vérifier la signature numérique de l'image ISO. L'empreinte observée par les auteurs de ce guide est, en admettant que c'est un exemplaire original que l'on a entre les mains :

```
0D24 B36A A9A2 A651 7878  7645 1202 821C BE2C D9C1
```

14.2.3 Installer Tails sur le support choisi

Retourner à la documentation officielle de Tails, disponible sur le site web section « Installer ou mettre à jour Tails » https://tails.boum.org/download/index.fr.html#index5h1.

Le fichier téléchargé est une « image ISO », c'est-à-dire un format de fichiers que la plupart des logiciels de gravure reconnaissent comme « image CD brute ». En général, si on insère un disque vierge ou réinscriptible dans son lecteur, qu'on fait un clic-droit sur le fichier téléchargé et qu'on choisit *Ouvrir avec Gravure de disque Brasero*, le logiciel de gravure s'occupe, après confirmation, de transformer cette image en l'écrivant sur le DVD.

14.3 Cloner ou mettre à jour une clé Tails

Une fois que l'on dispose d'un DVD, d'une clé USB ou d'une carte SD de Tails, il est possible de le dupliquer, par exemple pour créer une clé USB avec la persistance correspondant à une nouvelle identité contextuelle, pour donner une clé Tails à un ami, ou encore pour mettre à jour une clé USB contenant une ancienne version de Tails.

page ci-contre

Pour ce faire, on va suivre la documentation officielle de Tails, qui est disponible à partir de n'importe quel DVD, clé USB ou carte SD de Tails, même sans connexion à Internet.

plus bas

Démarrer Tails. Sur le bureau, cliquer sur l'icône *Documentation de Tails*. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Premier pas avec Tails* et cliquer sur la page *Installation sur une clé USB ou une carte SD*. C'est celle-ci qu'il s'agira de suivre.

Pour mettre à jour la clé ainsi créée, il faudra par la suite suivre la page *Mettre à jour une Clé USB Tails ou une carte SD Tails* dans la même section.

14.4 Démarrer sur un système *live*

Dès que la copie ou la gravure est terminée, on peut redémarrer l'ordinateur en laissant le support du système *live* dedans, et vérifier que la copie a fonctionné… à condition bien sûr qu'on ait configuré le micrologiciel pour qu'il démarre sur le bon support : voir la recette expliquant comment démarrer sur un média externe pour les détails.

page 93

Au démarrage, Tails affiche un écran qui permet de choisir, entre autres options, la langue d'affichage et la disposition du clavier.

14.5 Utiliser la persistance de Tails

Lorsqu'on utilise Tails à partir d'une clé USB ou d'une carte SD, il est possible de créer un volume persistant chiffré sur l'espace libre de la clé créé par l'Installeur de Tails. plus bas

Les données contenues dans le volume persistant sont sauvegardées et restent disponibles d'une session de travail à l'autre. Le volume persistant permet de sauvegarder des fichiers personnels, des clés de chiffrement, des configurations ou des logiciels qui ne sont pas installés par défault dans Tails.

plus bas Une fois le volume persistant créé, on peut choisir de l'activer, ou non, à chaque démarrage de Tails.

plus bas On pourra enfin le supprimer lorsqu'on ne voudra plus pouvoir accéder à ses données.

L'utilisation d'un volume persistant n'est toutefois pas sans conséquences quant aux traces laissées. C'est pourquoi il faudra commencer par lire la page d'avertissements concernant l'usage de la persistance.

Pour cela, cliquer sur l'icône *Documentation de Tails* se trouvant sur bureau. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Premier pas avec Tails* et cliquer sur la page *Avertissement à propos de la persistance*.

14.5.1 Créer et configurer un volume persistant

L'objectif de cette recette est de créer et de configurer un volume persistant sur une clé USB ou une carte SD de Tails.

Pour ce faire, on va suivre la documentation officielle de Tails, qui est disponible à partir de n'importe quel clé USB, carte SD ou DVD de Tails, même sans connexion à Internet. page préc.

Démarrer Tails. Sur le bureau, cliquer sur l'icône *Documentation de Tails*. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Premier pas avec Tails* et cliquer sur la page *Persistance* et enfin *Créer et configurer un volume persistant*. Suivre cette page de documentation.

Si l'on a déjà un volume persistant et que l'on souhaite simplement modifier ses paramètres, aller directement à la section *Options du volume persistant*.

14.5.2 Activer et utiliser un volume persistant

L'objectif de cette recette est d'activer le volume persistant nouvellement créé sur une clé USB ou une carte SD de Tails.

Pour ce faire, on va suivre la documentation officielle de Tails, qui est disponible à partir de n'importe quel clé USB, carte SD ou DVD de Tails, même sans connexion à Internet. page préc.

Démarrer Tails. Sur le bureau, cliquer sur l'icône *Documentation de Tails*. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Premier pas avec Tails* et cliquer sur la page *Persistance* et enfin *Activer et utiliser le volume persistant* et suivre cette page de documentation.

14.5.3 Supprimer un volume persistant

L'objectif de cette recette est de supprimer un volume persistant créé précédemment sur une clé USB ou une carte SD de Tails.

Pour ce faire, on va suivre la documentation officielle de Tails, qui est disponible à partir de n'importe quel clé USB, carte SD ou DVD de Tails, même sans connexion à Internet. page préc.

Démarrer Tails. Sur le bureau, cliquer sur l'icône *Documentation de Tails*. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Premier pas avec Tails* et cliquer sur la page *Persistance* et enfin *Supprimer le volume persistant* et suivre cette page de documentation.

14.5.4 Installer un logiciel additionnel persistant dans Tails

Tails contient des logiciels adaptés à la plupart des tâches courantes lors de l'utilisation d'Internet et de la création de documents. Toutefois, pour des projets spécifiques, on peut avoir besoin d'installer un logiciel spécifique dans *Tails*, comme par exemple un logiciel de conception et de simulation de circuits électroniques.

Lorsque *Tails* est installé sur une clé USB ou sur une carte SD, il est possible de configurer un volume persistant pour qu'un ou plusieurs logiciels spécifiques soient installés de manière automatique à chaque démarrage.

Trouver le nom de paquet à installer On a besoin du nom exact du paquet à installer. Pour le trouver, suivre la recette trouver un logiciel. Par exemple, notre logiciel de conception de circuits électroniques est fourni par le paquet `geda`.

page 116

Configurer les logiciels additionnels Si l'on ne dispose pas encore d'une clé USB ou d'une carte SD contenant un volume persistant, il faut en créer une grâce aux recettes cloner une clé *Tails* puis créer et configurer un volume persistant dans *Tails*.

page 101

On suivra ensuite l'outil créer et configurer un volume persistant dans *Tails* pour activer les options *Paquets APT* et *Listes APT*.

page ci-contre

page ci-contre

Redémarrer alors *Tails*. Une fois redémarré, au niveau de l'écran d'accueil, après avoir choisi le français comme langue cliquer sur *Plus d'option ?* → *Oui*. Puis activez la persistance. Maintenant cliquer sur *Suivant*. Choisir un *Mot de passe d'administration*, le rentrer deux fois et cliquer sur *Connexion*.

page ci-contre

Une fois sur le bureau, ouvrir un *Terminal administrateur* à partir du menu *Applications* → *Accessoires*. Entrer le mot de passe choisi précédemment puis *Valider*. Le terminal administrateur s'ouvre, taper dedans :

```
gedit /live/persistence/TailsData_unlocked/live-additional-software.conf
```

Puis taper sur *Entrer*.

Un fichier texte s'ouvre. Chaque ligne de ce fichier doit contenir le nom exact d'un paquet à installer. On va donc ajouter une ligne avec le nom du paquet trouvé précédemment. Dans notre exemple, on ajoutera la ligne :

```
geda
```

Enregistrer le fichier en cliquant sur *Enregistrer*. On peut maintenant fermer le fichier texte ainsi que le terminal et redémarrer *Tails*. Une fois la persistance activée et l'ordinateur connecté au réseau, notre logiciel devrait s'installer automatiquement au bout d'un temps plus ou moins long en fonction de la taille et du nombre de logiciels à installer (une fenêtre signale alors le succès de l'installation).

page ci-contre

Chapitre 15

Installer un système chiffré

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : compter une journée, avec plusieurs périodes d'attente (parfois longues).

On a vu que tout ordinateur — hormis avec certains systèmes *live* — laisse un peu partout des traces des fichiers ouverts, des travaux effectués, des connexions Internet, *etc.* On a vu aussi qu'une façon d'exposer un peu moins les données conservées sur l'ordinateur ainsi que les traces qu'on y laisse est de chiffrer le système sur lequel on travaille dans son ensemble.

page 19
page 37

Il est possible d'installer un système d'exploitation GNU/Linux comme Debian [1], sur une partie chiffrée du disque dur. À chaque démarrage, l'ordinateur va demander une phrase de passe, après quoi il débloque le chiffrement du disque, ce qui donne accès aux données, et permet donc le démarrage du système. Sans cette phrase, toute personne qui voudrait consulter le contenu de ce disque se trouvera face à des données indéchiffrables. C'est ce qu'on se propose de faire dans cette recette.

page 15

15.1 Limites

Attention ! Cette simple installation chiffrée ne règle pas tous les problèmes de confidentialité d'un coup de baguette magique. Elle ne protège les données qu'à certaines conditions.

15.1.1 Limites d'un système chiffré

Nous recommandons chaudement les lectures préalables suivantes :

- le chapitre concernant le chiffrement (et ses limites),
- le cas d'usage un nouveau départ, qui étudie, en détails, les limites pratiques d'un tel système et les attaques possibles contre lui.

page 37
page 57

Sans cela, l'installation d'un système chiffré peut procurer un sentiment erroné de sécurité, source de bien des problèmes.

15.1.2 Limites d'une nouvelle installation

Lors de l'installation d'un nouveau système, on part de zéro. Il n'y a aucun moyen simple de vérifier que le CD d'installation qu'on utilise est fiable, et ne contient pas

1. Pour chiffrer le disque dur lors de l'installation d'Ubuntu, il est nécessaire d'utiliser le CD nommé *alternate installer* [http://www.ubuntu.com/download/ubuntu/alternative-download#alternate].

par exemple de logiciels malveillants. On ne pourra éventuellement s'en rendre compte que *par la suite* — et peut-être qu'il sera trop tard…

15.1.3 Limites dans la prise en charge du matériel

Utiliser un système d'exploitation libre comme Debian a un désavantage : les fabriquants de matériel y font généralement peu attention. Il arrive donc qu'il ne soit pas facile, voire complètement impossible, d'utiliser un ordinateur ou l'un de ses périphériques avec Debian.

page 13

La situation s'améliore depuis quelques années : le fonctionnement du matériel tend à s'homogénéiser, et surtout, la diffusion des systèmes libres pousse de plus en plus les fabriquants à aider, directement ou non, à ce que leur matériel fonctionne [2].

Cependant, avant de remplacer un système d'exploitation, cela peut être une bonne idée de s'assurer que le matériel nécessaire fonctionne bien à l'aide d'un système *live*. Le système Tails, par exemple, est basé sur Debian. Le matériel qui fonctionne avec l'un devrait donc fonctionner avec l'autre sans difficultés.

page 99

15.2 Télécharger un support d'installation

Pour réaliser l'installation du système, le plus simple est d'utiliser un CD, un DVD ou une clé USB. Debian en propose plusieurs variantes, et il est donc nécessaire de commencer par choisir la méthode qui convient le mieux à notre situation.

15.2.1 Le CD d'installation par le réseau

Le plus rapide est d'utiliser un CD d'installation par le réseau. Le CD contient uniquement les tout premiers morceaux du système. Il télécharge ensuite, depuis Internet, les logiciels à installer. Il faut donc que l'ordinateur sur lequel on souhaite installer Debian soit connecté à Internet, de préférence par un câble réseau (et non par le *Wi-Fi* qui ne fonctionnera que rarement à l'intérieur du logiciel d'installation).

Il existe plusieurs fichiers (également appelés « images ») contenant une copie du CD d'installation, selon l'architecture du processeur. Dans la plupart des cas, il faudra télécharger celui dont le nom se termine par `amd64-i386-netinst.iso`, dit multi-architecture [3], qui conviendra pour les architectures 32 et 64 bits.

page 10

Cette image fonctionnera sur la plupart des ordinateurs domestiques fabriqués après 2006 [4].

15.2.2 Le DVD avec l'environnement graphique

S'il n'est pas possible de connecter à Internet l'ordinateur sur lequel on souhaite installer Debian, il est possible de télécharger un DVD contenant tout le système de base ainsi que l'environnement graphique habituel. Cela nécessite d'avoir accès à un graveur de DVD ou à une clé USB d'au moins 4GB.

2. Pour certains matériels, des problèmes peuvent venir de défauts dans le fonctionnement des microcodes intégrés. Ces problèmes sont parfois corrigés par des mises à jour que fournissent les fabriquants. Cela peut donc être une bonne idée de faire les mises à jour du micrologiciel (BIOS ou UEFI), de l'*Embedded Controller* ou d'autres composants avant de procéder à l'installation. Malheureusement, ces procédures diffèrent trop d'un matériel à un autre pour être détaillées dans cet ouvrage, mais peuvent en général être trouvées sur le site du constructeur…

3. Les images multi-architectures d'installation par le réseau sur le site de Debian [`http://cdimage.debian.org/debian-cd/current/multi-arch/iso-cd/`]

4. Pour les « vieux » Macs (*iBook G4* par exemple), il est nécessaire d'utiliser l'image dont le nom se termine par `powerpc-netinst.iso` que l'on trouvera sur le même site [`http://cdimage.debian.org/debian-cd/current/powerpc/iso-cd/`]. À noter : il n'est pas possible de créer une clé USB à partir de cette image, il vous faudra donc utiliser un DVD.

De la même manière que pour l'installation par le réseau, on choisira le DVD correspondant à notre architecture. Le DVD d'installation multi-architecture[5] devrait convenir pour la plupart des ordinateurs.

Seul le premier DVD est nécessaire pour réaliser l'installation. Le nom du fichier à télécharger devrait ressembler `debian-7.4.0-i386-amd64-source-DVD-1.iso`.

15.2.3 Utiliser une clé USB

Le système d'installation de Debian permet de transférer le contenu du CD d'installation par le réseau, ou celui du DVD, sur une clé USB dédiée à cela. On peut ainsi installer Debian sur un ordinateur n'ayant pas de lecteur CD ou DVD.

Transférer le DVD demande une clé d'une capacité de plus de 5 Go. Le CD d'installation par le réseau nécessite quant à lui une capacité de 512 Mo. Attention, cette opération nécessite l'utilisation de l'intégralité de la clé USB. Il faudra donc veiller à sauvegarder auparavant les fichiers qui s'y trouvent.

15.3 Vérifier l'empreinte du support d'installation

Il est bon de s'assurer que le téléchargement de l'image s'est bien déroulé en vérifiant l'empreinte de l'installeur, pour s'assurer de son intégrité et de son authenticité. Nous allons procéder en deux étapes, une première nous assurant de son intégrité, et une seconde assurant son authenticité. page 37

Pour cela, il est nécessaire de démarrer sur un système déjà installé. Si l'on a accès à un ordinateur sous GNU/Linux, tout va bien. Si on ne dispose que d'un système *live*, il est par exemple possible de mettre l'image téléchargée sur une clé USB, puis de vérifier l'empreinte à partir du système *live*. page 99

15.3.1 Vérifier l'intégrité du support d'installation

Pour cela suivre l'outil concernant les sommes de contrôle. Il sera nécessaire de calculer la somme de contrôle SHA512 du support d'installation téléchargé et vérifier si celle-ci correspond à celle contenue dans le fichier `SHA512SUMS` contenu dans le même dossier que le support d'installation. Fichier que l'on devra télécharger également. page 153

15.3.2 Vérifier l'authenticité du support d'installation

Si la vérification de l'intégrité s'est bien déroulée, à savoir si les deux sommes de contrôles calculées correspondent, on peut poursuivre le processus afin de vérifier son authenticité. En effet, un adversaire pourrait fournir un support d'installation corrompu et sa somme de contrôle. La vérification précédente nous permettrait simplement de constater que le fichier téléchargé est bien celui qui était disponible sur le site web, pas qu'il est celui qu'on espère avoir. Le deuxième tome explique comment s'assurer de l'authenticité de l'installeur téléchargé, car l'empreinte est signée avec GnuPG, qui utilise la cryptographie asymétrique, il faudra donc directement se référer à l'outil concernant la vérification d'une signature. Après avoir téléchargé le fichier `SHA1SUMS.sign`, suivre les étapes permettant de vérifier la signature cryptographique du fichier `SHA512SUMS`. tome 2 § 18.8

15.4 Préparer les supports d'installation

Une fois l'image du support d'installation choisie et téléchargée et vérifiée, il nous reste à la transférer sur un CD, un DVD ou une clé USB.

5. Les DVD d'installation de Debian multi-architecture [http://cdimage.debian.org/debian-cd/current/multi-arch/iso-dvd/]

15.4.1 Graver le CD ou le DVD d'installation

Le fichier téléchargé est une « image ISO », c'est-à-dire un format de fichiers que la plupart des logiciels de gravure reconnaissent comme « image CD brute ». En général, si on insère un disque vierge dans son lecteur, qu'on fait un clic-droit sur le fichier et qu'on choisit *Gravure de disque Brasero*, le logiciel de gravure s'occupe tout seul de transformer cette image en l'écrivant sur le disque vierge — en tout cas, ça marche avec Tails, et plus généralement sous Debian ou Ubuntu.

Sous Windows, si on a pas déjà installé de logiciel capable de graver des images ISO, le logiciel libre InfraRecorder[6] fera parfaitement l'affaire.

15.4.2 Créer une clé USB d'installation

page 99

Pour créer une clé USB d'installation, il est nécessaire d'effectuer les opérations qui suivent à partir d'un système basé sur Linux tel que Debian ou Tails.

Se munir d'une clé USB vierge, ou contenant uniquement des données auxquelles on ne tient pas.

page 32

Attention : les données éventuellement présentes sur la clé seront perdues. Par contre, il serait facile de procéder à une analyse pour retrouver les fichiers dont le contenu n'aurait pas été écrasé auparavant…

Repérer l'emplacement de la clé USB

Pour la suite des opérations, il est nécessaire de déterminer le nom que le système attribue à la clé USB. Pour ce faire, on va démarrer l'*Utilitaire de disque* dans le menu *Applications* → *Accessoires*.

Une fois ce dernier ouvert, on peut brancher notre clé USB. Une entrée correspondant à cette dernière devrait apparaître dans la liste située à gauche. Après l'avoir sélectionnée, on pourra lire le *nom* du disque dans la partie à droite, sous l'intitulé *Disque*, à côté de l'étiquette *Périphérique*.

Cela doit ressembler à `/dev/sdx` : c'est ainsi que l'ordinateur a identifié le support de stockage que l'on vient de brancher. C'est le nom qu'on devra écrire par la suite dans certaines commandes, à la place de `LE_PÉRIPHERIQUE`.

Comme on va remplacer ce qui se trouvait sur la clé, il faut également demander au système de ne plus utiliser ce qu'elle contient : toujours dans la partie droite, on sélectionne donc, tour à tour, chacun des *Volumes* dessinés. Pour chacun d'entre eux, on clique sur *Démonter le volume* si ce bouton est disponible. S'il s'agit d'une partition chiffrée, il nous faudra également cliquer juste au dessus, sur la partie intitulée "Chiffré", puis sur le bouton *Verrouiller le volume*.

Plutôt que de fermer l'*Utilitaire de disque* dès maintenant, mieux vaut réduire sa fenêtre : on en aura encore besoin une dernière fois après la copie proprement dite.

Lancer la copie brute

page 85

On va maintenant ouvrir un *Terminal*, tout en gardant à portée de souris l'icône de l'image ISO téléchargée auparavant.

On va commencer la commande en tapant (**sans** faire *Entrée*) :

```
cat
```

6. http://infrarecorder.org/

Ajouter ensuite un espace. Puis, on va indiquer la *source* de la copie (la chose que l'on va copier). Pour ce faire, il faut, avec la souris, cliquer sur l'icône du fichier ISO que l'on vient de télécharger puis de vérifier et le faire glisser dans la fenêtre du terminal. Après avoir relâché le bouton, ce qui est affiché doit ressembler à :

```
cat '/home/domi/Desktop/debian-7.4.0-amd64-i386-netinst.iso'
```

Ce n'est toujours pas fini, car il faut maintenant indiquer la *destination* de la copie, en ajoutant à la fin de notre commande :

```
> LE_PÉRIPHÉRIQUE
```

Une fois cela fait, la commande complète doit ressembler à quelque chose comme :

```
cat '/home/domi/Desktop/debian-7.4.0-amd64-i386-netinst.iso' > /dev/sdx
```

La copie se lance dès qu'on a appuyé sur *Entrée*, ne laissant plus apparaître qu'un sobre carré à la ligne suivante.

Après de la patience et le retour du signe `#` symbolisant l'invite de commande, on peut fermer le terminal. Reste ensuite à faire réapparaître la fenêtre de l'*Utilitaire de disque* et à cliquer sur *Déconnexion en toute sécurité* pour être sûr de ne pas abîmer les données fraîchement copiées sur notre clé USB.

15.4.3 Mettre les « microcodes » sur une clé USB

Certains périphériques de l'ordinateur peuvent nécessiter, pour fonctionner, que le système leur fournisse un « microcode » (ou *firmware*)… mais ce n'est pas toujours le cas. page 13

On peut donc tenter de réaliser l'installation sans s'en préoccuper et revenir lire cette section plus tard si l'on rencontre des problèmes.

Un micro—quoi ?

Ces microcodes sont des programmes qui ont la particularité de s'exécuter sur des puces électroniques à l'intérieur du périphérique et non sur le processeur de l'ordinateur. C'est par exemple le cas du programme qui contrôlera le déplacement des parties mécaniques d'un disque dur ou le fonctionnement du système de radio d'une carte Wi-Fi. On ne se rend pas forcément compte qu'ils existent… car la plupart des microcodes sont livrés directement avec le matériel. page 10

Mais pour d'autres périphériques, le système d'exploitation doit envoyer le microcode à un composant lors de son initialisation.

Ceux qui sont libres sont livrés avec le programme d'installation de Debian. Malheureusement, la plupart des microcodes ne sont pas libres. Nous devons donc mettre nous-mêmes à disposition du programme d'installation tout microcode non-libre nécessaire au fonctionnement de l'ordinateur : c'est typiquement le cas pour certaines cartes Wi-Fi. page 29

Même si c'est hautement improbable, on peut envisager que le microcode propriétaire d'une carte Wi-Fi nous espionne à notre insu… sauf que sans microcode, elle ne fonctionnera tout simplement pas. C'est encore une fois une histoire de compromis.

Obtenir les microcodes supplémentaires

Une bonne partie des microcodes non-libres sont néanmoins redistribués par le projet Debian.

Ils sont livrés dans une archive nommée `firmware.tar.gz`, que l'on peut télécharger sur la page : `http://cdimage.debian.org/cdimage/unofficial/non-free/firmware/wheezy/current/`

Utiliser une clé USB en plus d'un CD ou d'un DVD

Si on utilise un CD ou un DVD pour réaliser l'installation, il est nécessaire de copier les microcodes supplémentaires sur une clé USB. Il est possible d'utiliser une clé USB contenant déjà des données, mais cette clé ne doit pas être chiffrée.

Pour cela, il faut créer dans la clé un nouveau répertoire que l'on baptisera `firmware`[7], et y extraire (avec le *Gestionnaire d'archives*) le contenu de l'archive que l'on vient de télécharger.

Pour ce faire, il suffit (avec un système GNU/Linux) de double-cliquer sur le fichier `firmware.tar.gz` qu'on vient de télécharger. Le *Gestionnaire d'archives* s'ouvre : cliquer sur le bouton *Extraire* dans le menu du haut, et choisir ensuite comme emplacement le dossier `firmware` de la clé USB en question. On peut ensuite démonter la clé.

Utiliser une seule clé USB

Si on réalise l'installation à partir d'une clé USB, il est possible de copier les microcodes sur cette même clé.

Pour cela, il faut commencer par ouvrir l'*Utilitaire de disque* à partir du menu *Applications* → *Accessoires*. Puis :

- brancher la clé USB dans l'ordinateur ;
- sélectionner la clé USB dans la liste de gauche ;
- dans la partie droite, cliquer sur la zone indiquée comme *Libre* sous l'intitulé *Volumes* ;
- cliquer sur le bouton *Créer une partition* en-dessous ;
- choisir `FAT` comme *Type* et `firmware` comme nom ;
- cliquer sur le bouton *Créer*.

On peut maintenant fermer l'*Utilitaire de disque*, utiliser le *Gestionnaire d'archives* pour extraire le contenu de l'archive téléchargée précédemment sur l'espace de la clé USB s'appelant *firmware*, et démonter cette dernière.

15.5 L'installation proprement dite

page 93

Pour installer la Debian chiffrée depuis le support d'installation (CD, DVD ou clé USB), il faut démarrer sur celui-ci en suivant la recette correspondante.

À partir de là, l'installation proprement dite peut commencer : prévoir du temps devant soi et quelques mots croisés, car l'ordinateur pourra travailler longtemps sans surveillance particulière.

Vérifier, dans le cas d'un CD d'installation par le réseau, que le câble reliant l'ordinateur au réseau est bien branché. Et s'il s'agit d'un ordinateur portable, vérifier que le câble d'alimentation est branché, car il n'y a pas de notification de batterie faible durant l'installation.

7. Le nom du répertoire doit *vraiment* être `firmware`, sinon cela ne marchera pas.

Le programme d'installation de Debian dispose de sa propre documentation [8]. En cas de doute à la lecture des étapes décrites par la suite, cela peut valoir le coup d'y jeter un œil. Par ailleurs, pour la plupart des choix qu'il nous demande de faire, le programme d'installation nous proposera automatiquement une réponse qui fonctionne généralement…

15.5.1 Lancement de l'installeur

On démarre donc sur le support d'installation (CD, DVD ou clé USB). Un premier menu nommé *Debian GNU/Linux installer boot menu* apparaît.

Dans le cas où on a choisi un CD multi-architecture, certaines options seront présentes en double, marquées comme « 64-bit ».

L'option sélectionnée automatiquement par l'installeur sera soit *Install*, soit *64-bit install* ; dans ce dernier cas, le programme d'installation a détecté que le processeur est compatible avec l'architecture `amd64`, qui apporte quelques avantages en termes de sécurité.

page 10

Il est plus commode d'utiliser la souris lors de l'installation ; on choisira donc *Graphical install*, dans le cas où *Install* était pré-sélectionné, et plutôt *64-bit graphical install*, si *64-bit install* avait été sélectionné tout seul. Ce choix s'effectue en se déplaçant avec les touches ↓ et ↑ du clavier.

Une fois la bonne ligne sélectionnée, il faut appuyer sur la touche *Entrée* pour lancer la suite du programme d'installation.

15.5.2 Choisir la langue et la disposition du clavier

- Après un peu de patience, un menu nommé *Select a language* apparaît : l'installeur propose de choisir une langue pour la suite de l'installation. Toujours en se déplaçant avec les flèches, sélectionner *Français* et appuyer sur la touche *Entrée*.
- Un menu demande le pays, pour peaufiner l'adaptation du système. Choisir son lieu géographique, et appuyer sur *Entrée*.
- Dans *Configurer le clavier*, le choix par défaut *Français* convient si l'on a un clavier français « azerty ».
- L'installeur charge ensuite les fichiers dont il a besoin.

15.5.3 Configuration du réseau et baptême de la machine

- L'installeur prend alors un peu de temps pour configurer le réseau. Si notre ordinateur possède plusieurs cartes réseau, il faut choisir celle dont on va se servir pour l'installation. Le choix par défaut est généralement le bon, il s'agit d'une carte réseau *Ethernet*. On nous demande ensuite le *Nom de machine*. Choisir un petit nom pour son ordinateur, en sachant que ce nom sera ensuite visible depuis le réseau, et pourra aussi s'inscrire dans les fichiers créés ou modifiés avec le système qu'on est en train d'installer.
- L'installeur demande le *Domaine*. Sans entrer dans les détails, mieux vaut laisser ce champ vide (donc effacer ce que le programme peut éventuellement avoir prérempli).

15.5.4 Créer les utilisateurs et choisir les mots de passe

Le programme d'installation nous demande maintenant de choisir le *mot de passe du superutilisateur*. C'est ce mot de passe qui sera nécessaire pour réaliser les tâches

8. Le manuel d'installation est disponible dans de nombreuses versions [`http://www.debian.org/releases/wheezy/installmanual.fr.html`]. On suivra celui correspondant à l'architecture du processeur [page 10].

d'administration de l'ordinateur : mises à jour, installation de logiciels, modifications majeures du système, *etc.*

Il est toutefois possible de s'épargner un mot de passe supplémentaire, et de permettre que le premier compte créé sur le système ait le droit de faire des opérations d'administration [9]. Pour cela, il suffit de ne pas entrer de mot de passe pour le superutilisateur : laisser simplement la case vide.

Cela dit, ce choix doit être bien pesé au préalable : souvent, il est plus simple d'utiliser cette méthode, notamment parce qu'il n'y a pas un mot de passe supplémentaire à retenir. Cependant, dans sa configuration par défaut, elle peut permettre à n'importe quel programme lancé dans ce compte, *sans que celui-ci nous demande confirmation au préalable*, d'effectuer des opérations en disposant des privilèges d'administrateur ; et ce, pendant quinze minutes après la saisie du mot de passe.

- Dans *Nom complet du nouvel utilisateur* choisir le nom associé au premier compte créé sur le système. Ce nom sera souvent enregistré dans les documents créés ou modifiés dans cette session ; il peut donc être intéressant de choisir un nouveau pseudonyme.
- Dans *Identifiant pour le compte utilisateur*, choisir un identifiant (*login*) pour ce compte. Il est prérempli, mais peut être modifié. L'installeur prévient, pour le cas où l'on voudrait le changer, qu'il doit commencer par une lettre minuscule et être suivi d'un nombre quelconque de chiffres et de lettres minuscules.
- L'installeur demande un mot de passe pour l'utilisateur qui aura le droit d'administrer l'ordinateur, si l'on a décidé de ne pas entrer un mot de passe « superutilisateur » précédemment.

15.5.5 Partitionner les disques

Le CD démarre ensuite l'outil de partitionnement. Il détecte les partitions présentes, et va proposer de les modifier.

- Dans le menu *Méthode de partitionnement*, choisir *Assisté — utiliser tout un disque avec LVM chiffré.*
- Dans *Disque à partitionner* choisir le disque sur lequel installer Debian GNU/Linux. Si l'on veut supprimer le système actuellement installé, il correspond en général au premier disque de la liste. La taille du disque est un indice permettant de ne pas se tromper, pour ne pas essayer d'installer Debian sur la clé USB contenant l'installeur par exemple.
- L'installeur propose ensuite différents *Schémas de partitionnement.* Là, il y a plusieurs possibilités :
  - *Tout dans une seule partition* fonctionne toujours ;
  - si l'on a un gros disque (pas moins de 20 Go), on peut décider de stocker le répertoire `/home`, qui contiendra vos données personnelles, dans une partition séparée.
- L'installeur prévient alors qu'il va appliquer le schéma actuel de partitionnement, ce qui sera irréversible. Vu que l'on a bien fait les sauvegardes de ce que l'on voulait garder, répondre *Oui* à *Écrire les modifications sur les disques et configurer LVM ?*
- L'installeur va alors remplacer l'ancien contenu du disque par des données aléatoires. C'est très long — de nombreuses heures sur un gros disque — et ça laisse donc du temps pour faire autre chose !
- L'installeur demande alors une *Phrase secrète de chiffrement.* Choisir une bonne phrase de passe et la taper. page 91
- Confirmer la phrase de passe en la tapant une seconde fois.
- L'installeur montre une liste de toutes les partitions qu'il va créer. Il est possible de lui faire confiance et de *Terminer le partitionnement et appliquer les changements.*

9. Ce mode est appelé *sudo*, car dans le terminal, il sera possible, en ajoutant `sudo` au début de la ligne, d'exécuter une commande en tant que « superutilisateur ».

- L'installeur prévient qu'il va détruire toutes les données présentes sur le disque. Tout le disque a déjà été rempli de données aléatoires, donc s'il contenait des données importantes elles ont déjà été effacées. Répondre *Oui* à *Faut-il appliquer les changements sur les disques ?* L'installeur crée alors les partitions, ce qui peut prendre un petit bout de temps.

15.5.6 Installation du système de base

L'installeur va maintenant installer un système GNU/Linux minimal. Le laisser faire…

15.5.7 Choix du serveur Debian

Si cette question n'apparaît pas à ce moment, pas d'inquiétude, c'est simplement que l'installeur utilisé n'est pas celui par le réseau. Dans ce cas, elle arrivera un peu plus tard au cours de l'installation.

- L'installeur demande de choisir le *Pays du miroir de l'archive Debian*. Le choix par défaut *France* est bon si l'on est en France.
- Il demande ensuite le *Miroir de l'archive Debian* à utiliser. Le choix par défaut *ftp.fr.debian.org* est aussi très bien.
- L'installeur demande si on a besoin d'un *Mandataire HTTP*. On laisse vide.
- L'installeur télécharge alors les fichiers dont il a besoin pour continuer.

15.5.8 Sélection des logiciels

La prochaine question concerne la *configuration de popularity-contest* et demande *Souhaitez-vous participer à l'étude statistique sur l'utilisation des paquets ?* Il est possible de répondre *Oui* sans risque de divulguer beaucoup d'informations supplémentaires : vu que les logiciels seront de toute façon téléchargés à partir des serveurs de Debian, ceux-ci pourraient déjà savoir quels paquets on utilise s'ils le voulaient.

L'installeur demande quels sont les *Logiciels à installer*. Ses propositions conviennent en général : *environnement de bureau Debian*, *serveur d'impression* et *Utilitaires usuels du système*, plus *ordinateur portable* le cas échéant.

L'installeur télécharge alors tout le reste du système Debian GNU/Linux et l'installe. C'est long, il y a le temps d'aller faire autre chose.

15.5.9 Installation du programme de démarrage GRUB

L'installeur propose de mettre en place le programme de démarrage, qui permet de démarrer Linux, sur une partie du disque dur appelée « secteur d'amorçage ». L'installeur propose un *Périphérique où sera installé le programme de démarrage*, laisser le choix par défaut, cliquer sur *Continuer* et attendre un peu.

Lorsqu'il a terminé, l'installeur propose de redémarrer l'ordinateur en vérifiant que le support d'installation (CD, DVD, clé USB) n'est plus inséré lors du redémarrage. Choisir *Continuer*.

15.5.10 Redémarrer sur le nouveau système

L'ordinateur démarre alors sur le nouveau système. À un moment, il demande la phrase de passe sur un écran noir : « `Enter passphrase:` ». La taper, sans s'inquiéter que rien ne s'affiche, et appuyer sur la touche *Entrée* à la fin [10].

10. Si l'on n'est pas très à l'aise avec la frappe au clavier, il arrive souvent dans les premiers temps qu'on fasse une erreur de frappe dans la phrase, et c'est d'autant plus probable qu'aucun caractère ne s'affiche. Ne pas s'inquiéter des erreurs répétées, et insister jusqu'à réussir à taper la phrase sans faute… au bout de quelque temps, elle sera « rentrée dans les doigts », et les fautes de frappe se feront plus rares. Cela dit, il ne coûte rien de vérifier qu'on n'a pas malencontreusement laissé la

Après le démarrage d'un certain nombre de programmes, un écran apparaît avec le nom de la machine et le nom du compte utilisateur entré précedemment. Il faut sélectionner ce dernier, puis entrer le mot de passe associé.

Voilà un nouveau système Debian chiffré prêt à être utilisé. Pour qui n'en aurait jamais utilisé, se balader dedans peut être une bonne idée pour s'y familiariser. En haut de l'écran, le menu *Applications* permet d'accéder aux nombreux logiciels déjà installés. Les pages d'aide contenant de nombreux conseils et astuces sont accessibles à travers le menu *Applications* → *Accessoires* → *Aide*.

15.6 Quelques pistes pour continuer

page 143
page 127
page 169

Il peut maintenant être utile d'apprendre à sauvegarder des données… et à en effacer « pour de vrai ».

Il est également important d'apprendre à garder son système à jour. Des problèmes affectant les logiciels sont découverts régulièrement, et il est important d'installer les corrections au fur et à mesure de leur disponibilité.

15.7 Un peu de documentation sur Debian et GNU/Linux

Voici quelques références de documentations sur Debian et GNU/Linux :

- Le guide de référence officiel de Debian [11] ;
- La page d'accueil de la documentation officielle d'utilisation de Debian [12] ;
- La Formation Debian GNU/Linux [13] : une excellente auto-formation sur Debian en français ;
- Le cahier de l'administrateur Debian [14].

On peut trouver beaucoup de documentation sur l'utilisation de GNU/Linux. Si elles sont souvent très utiles, elles sont malheureusement, comme beaucoup de choses sur Internet du reste, de qualité inégale. En particulier, beaucoup d'entre elles arrêteront de fonctionner lorsqu'une partie du système sera modifiée, ou seront peu soucieuses de l'intimité que l'on attend de notre système. Il faut donc faire preuve d'esprit critique et tenter de les comprendre avant de les appliquer.

Ceci dit, voici encore quelques références de wikis et des forums :

- Le wiki officiel de Debian [15] (partiellement traduit de l'anglais) ;
- Le forum en français sur Debian `debian-fr.org` [16] ;
- Andesi [17] : un wiki et forum en français sur Debian.

touche VerrMaj enfoncée, auquel cas l'on pourrait s'acharner longtemps sur le clavier sans pour autant arriver à déverrouiller le disque dur.

11. http://www.debian.org/doc/manuals/debian-reference/index.fr.html
12. http://www.debian.org/doc/user-manuals
13. http://formation-debian.via.ecp.fr/
14. http://debian-handbook.info/browse/fr-FR/stable/
15. http://wiki.debian.org/
16. http://forum.debian-fr.org/
17. http://www.andesi.org/

Chapitre 16

Choisir, vérifier et installer un logiciel

Cette partie propose quelques recettes à propos de la gestion de ses logiciels :

- Comment trouver un paquet Debian ? Lorsqu'on cherche à réaliser de nouvelles tâches avec un ordinateur, on est souvent amené à installer de nouveaux logiciels… quelques conseils pour trouver ce que l'on cherche dans Debian ; page suivante
- Avec quels critères de choix ? On doit parfois choisir un logiciel pour effectuer une certaine tâche, et il est alors courant de se sentir perdu dans la multitude de solutions disponibles… quelques critères permettant de prendre une décision adéquate ; page 117
- Comment installer un paquet Debian ? Une fois que l'on sait quel paquet contient le logiciel que l'on veut utiliser, reste à l'installer proprement ; page 120
- Comment modifier ses dépôts Debian ? Les paquets Debian qui contiennent les programmes se trouvent dans ce qu'on appelle des *dépôts*. Si les dépôts fournis avec Debian contiennent quasiment tous les logiciels dont on peut avoir besoin, il est parfois utile d'ajouter de nouveaux dépôts. page 122

16.1 Trouver un logiciel

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : de 5 minutes (si l'on connaît le nom du logiciel que l'on cherche) à une demi-heure (si l'on part de zéro).

Parfois, on connaît déjà le nom du logiciel que l'on souhaite installer — parce qu'on nous l'a conseillé, parce qu'on l'a trouvé sur Internet — et l'on veut savoir s'il est dans Debian. D'autres fois, on connaît seulement la tâche que l'on souhaiterait que le logiciel remplisse. Dans tous les cas, la base de données des logiciels disponibles dans Debian répondra certainement à nos questions.

page ci-contre

Pour faire des choix éclairés, lorsque plusieurs logiciels permettent d'effectuer une même tâche, voir choisir un logiciel.

- Ouvrir le *gestionnaire de paquets Synaptic* à partir du menu *Applications* → *Outils système* → *Administration*. Puisque le gestionnaire de paquets permet de modifier les logiciels installés sur l'ordinateur, et donc de choisir à quels programmes on fait confiance, on est rassuré qu'il nous demande notre mot de passe pour s'ouvrir.
- Une fois dans le gestionnaire de paquets, commençons par recharger la liste des paquets disponibles en cliquant sur l'icône *Recharger*. Le gestionnaire de paquets télécharge alors les dernières informations sur les paquets disponibles depuis un serveur Debian.
- Ensuite, il y a deux techniques pour chercher un paquet :
  - soit cliquer sur l'icône *Rechercher* dans la barre d'outils. Là, vérifier que *Description et nom* est bien sélectionné dans *Rechercher dans*. Taper des mots-clé ou un nom d'application dans la case *Rechercher* (par exemple « dictionnaire allemand openoffice »). Les descriptions des applications peu courantes sont rarement traduites en français. Avec quelques bases d'anglais, il est souvent intéressant d'essayer des mots-clé dans cette langue. Cliquer sur *Rechercher* pour lancer la recherche ;
  - soit sélectionner une catégorie dans la colonne de gauche.
- Les résultats de la recherche ou les paquets de la catégorie s'affichent alors dans la liste en haut à droite. En cliquant sur le nom d'un paquet, sa description apparaît dans le cadre en bas à droite.
- Reste maintenant à installer le paquet correspondant.

page 120

16.2 Critères de choix

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : une demi-heure à une heure.

On est parfois amené à choisir un logiciel pour effectuer une certaine tâche, et il est alors courant de se sentir perdu dans la multitude de solutions disponibles. Voici donc quelques critères permettant de prendre une décision adéquate.

L'intérêt d'utiliser des logiciels libres par rapport à des logiciels propriétaires ou *open source* a d'ores et déjà été expliqué. La suite du texte s'attachera donc uniquement à départager les logiciels libres disponibles.

page 29

16.2.1 Mode d'installation

Il est généralement préférable d'installer des logiciels fournis par sa distribution GNU/Linux (par exemple, Debian). Il y a deux principales raisons à ça.

Tout d'abord, une question pratique : la distribution fournit les outils pour installer et mettre à jour, de façon plus ou moins automatisée, un ensemble de logiciels ; elle nous alerte lorsqu'une faille de sécurité affecte l'un des logiciels que l'on utilise. Mais dès lors qu'on installe un logiciel qui n'est pas fourni par sa distribution, on est livré à soi-même : il faut penser à le mettre à jour, se tenir informé des failles de sécurité qui y sont découvertes, gérer les dépendances entre logiciels. Ça demande des efforts, du temps, des compétences.

D'autre part, une question de politique de sécurité : lorsqu'on a choisi sa distribution GNU/Linux, on a implicitement décidé d'accorder une certaine confiance à un ensemble de gens, à un processus de production. Installer un logiciel qui n'est pas fourni par sa distribution implique de prendre une décision similaire à propos d'un nouvel ensemble de gens, d'un nouveau processus de production. Une telle décision ne se prend pas à la légère : lorsqu'on décide d'installer un logiciel n'appartenant pas à sa distribution, on élargit l'ensemble des personnes et processus à qui on accorde de la confiance, et on augmente donc les risques.

16.2.2 Maturité

L'attrait de la nouveauté qui lave plus blanc que blanc est bien souvent un piège.

Mieux vaut, autant que possible, choisir un logiciel ayant atteint une certaine maturité : dans un logiciel activement développé et utilisé depuis au moins quelques années, il y a des chances que les plus gros problèmes aient déjà été découverts et corrigés… y compris les failles de sécurité.

Pour s'en rendre compte, il faut consulter l'historique de chacun des logiciels, sur leur site web ou dans le fichier nommé `Changelog` (ou approchant), généralement livré avec le logiciel.

16.2.3 Processus de production et « communauté »

L'étiquette *logiciel libre* est un critère essentiellement juridique, qui ne doit jamais suffire à nous inspirer confiance.

page 30

Certes, le fait qu'un logiciel soit placé sous une licence libre ouvre la possibilité de modes de développement inspirant confiance.

Mais les personnes développant ce logiciel peuvent fort bien, intentionnellement ou non, décourager toute coopération et travailler en vase clos. Que nous importe alors

que le programme soit *juridiquement* libre, si, de fait, personne d'autre ne lira jamais son code source ?

Il convient donc d'étudier rapidement le processus de production des logiciels en lice, en s'aidant des questions suivantes, qui nous permettront de surcroît de jauger le dynamisme du processus :

- Qui développe ? Une personne, des personnes, toute une équipe ?
- Le nombre de personnes qui contribuent au code source va-t-il en augmentant ou en diminuant ?
- Le développement est-il actif ? Il ne s'agit pas ici de vitesse pure, mais de réactivité, de suivi à long terme, de résistance. Le développement logiciel est une course d'endurance et non un *sprint*.

Et à propos des outils de communication collective sur lesquels s'appuie le développement (listes et salons de discussion, par exemple) :

- A-t-on facilement accès aux discussions guidant le développement du logiciel ?
- Ces discussions rassemblent-elles de nombreuses personnes ?
- Ces personnes participent-elle à son développement, ou ne font-elles que l'utiliser ?
- Quelle atmosphère y règne ? Calme plat, silence de mort, joyeuse cacophonie, sérieux glaçant, bras ouverts, hostilité implicite, tendre complicité, *etc.* ?
- Le volume de discussion, sur les derniers mois/années, va-t-il en diminuant ou en augmentant ? Plus que le volume brut, c'est surtout la proportion de messages obtenant des réponses qui importe : un logiciel mûr, stable et bien documenté ne sera pas forcément source de discussions, mais si plus personne n'est là pour répondre aux questions des néophytes, ça peut être mauvais signe.
- Peut-on trouver des retours d'utilisation, des suggestions d'améliorations ? Si oui, sont-elles prises en compte ?
- Les réponses sont-elles toujours données par un nombre réduit de personnes, ou existe-t-il des pratiques d'entraide plus large ?

16.2.4 Popularité

La popularité est un critère délicat en matière de logiciels. Le fait que la grande majorité des ordinateurs de bureau fonctionnent actuellement sous Windows n'indique en rien que Windows soit le meilleur système d'exploitation disponible.

Pour autant, si ce logiciel n'est pas utilisé par beaucoup de monde, on peut douter de sa viabilité à long terme : si l'équipe de développement venait à cesser de travailler sur ce logiciel, que deviendrait-il ? Qui reprendrait le flambeau ?

On peut donc retenir, comme règle générale, qu'il faut choisir un logiciel utilisé par un nombre suffisamment important de personnes, mais pas forcément *le* logiciel le plus utilisé.

Afin de mesurer la popularité d'un logiciel, il est possible, d'une part, d'utiliser les mêmes critères que ceux décrits ci-dessus au sujet du dynamisme de la « communauté » formée autour de lui. D'autre part, Debian publie les résultats de son concours de popularité [1], qui permet de comparer non seulement le nombre de personnes ayant installé tel ou tel logiciel, mais aussi, voire surtout, l'évolution dans le temps de leur popularité.

16.2.5 Passé de sécurité

Voici de nouveau un critère à double tranchant.

1. Debian.org, 2014, *Debian Popularity Contest* [http://popcon.debian.org/] (en anglais).

On peut commencer par jeter un œil sur le suivi de sécurité [2] proposé par Debian. En y cherchant un logiciel par son nom, on peut avoir la liste des problèmes de sécurité qui y ont été découverts et parfois résolus.

Si ce logiciel a un historique de sécurité parfaitement vierge, ça peut impliquer soit que tout le monde s'en fout, soit que le logiciel est écrit de façon extrêmement rigoureuse.

Si des failles de sécurité ont été découvertes dans le logiciel étudié, il y a plusieurs implications, parfois contradictoires.

1. Ces failles ont été découvertes et corrigées :
   - donc elles n'existent plus, *a priori* ;
   - donc quelqu'un s'est préoccupé de les trouver, et quelqu'un d'autre de les corriger : on peut supposer qu'une attention est donnée à cette question.
2. Ces failles ont existé :
   - le logiciel est peut-être écrit sans que la sécurité soit un souci particulier ;
   - d'autres failles peuvent subsister, non encore découvertes ou pire, non encore publiées.

Afin d'affiner notre intuition par rapport à ce logiciel, il peut être bon de se pencher sur le critère « temps » : par exemple, il n'est pas dramatique que quelques failles aient été découvertes au début du développement d'un logiciel, si aucune n'a été découverte depuis quelques années ; on peut alors mettre ça sur le compte des erreurs de jeunesse. Au contraire, si de nouvelles failles sont découvertes régulièrement, depuis des années, et jusqu'à très récemment, il est fort possible que le logiciel ait encore de nombreux problèmes de sécurité totalement inconnus… ou non publiés.

Pour illustrer le propos, il est possible de comparer l'historique des failles de Claws Mail [3] et celui de Thunderbird [4].

16.2.6 Équipe de développement

Qui a écrit, qui écrit ce logiciel ? Si l'on a réussi à répondre à cette question, divers indices peuvent nous aider à déterminer la confiance qui peut être accordée à l'équipe de développement. Par exemple :

- Les mêmes personnes ont aussi écrit un autre logiciel, que nous utilisons déjà intensivement ; nos impressions sur cet autre logiciel sont tout à fait pertinentes dans le cadre de cette étude.
- Des membres de l'équipe de développement ont des adresses qui finissent par `@debian.org`, et ont donc le droit de modifier les logiciels fournis par Debian GNU/Linux ; si nous utilisons cette distribution, nous accordons déjà, de fait, une certaine confiance à ces personnes.
- Des membres de l'équipe de développement ont des adresses qui finissent par `@google.com`, ce qui montre que Google les paie ; s'il n'y a aucun doute à avoir sur leurs compétences techniques, on peut se demander à quel point leur travail est téléguidé par leur employeur qui, lui, n'est digne d'aucune confiance quant à ses intentions concernant vos données personnelles.

2. L'équipe de sécurité de Debian maintient des informations pour chacun des paquets, visibles sur le *security tracker* [http://security-tracker.debian.net/tracker/].
3. http://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=claws+mail
4. http://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=thunderbird

16.3 Installer un paquet Debian

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 5 minutes, plus le temps de téléchargement et d'installation (quelques secondes à plusieurs heures selon la taille des logiciels à installer et la vitesse de la connexion).

16.3.1 Ouvrir le gestionnaire de paquets

Une fois que l'on sait quel paquet contient le logiciel que l'on veut utiliser, reste à l'installer. Pour cela, on va utiliser le *Gestionnaire de paquets Synaptic* que l'on peut ouvrir à partir du menu *Applications* → *Outils Système* → *Administration*.

Puisque le gestionnaire de paquets permet de modifier les logiciels installés sur l'ordinateur, un mot de passe est nécessaire pour l'ouvrir.

16.3.2 Recharger la liste des paquets disponibles

Une fois dans le gestionnaire de paquets, commençons par recharger la liste des paquets disponibles en cliquant sur l'icône *Recharger*. Le gestionnaire de paquets télécharge alors les dernières informations sur les paquets disponibles depuis les serveurs de Debian.

16.3.3 Rechercher le paquet à installer

Ensuite, on va trouver le paquet qu'on veut installer. On clique sur l'icône *Rechercher* dans la barre d'outils. Là, si on connait le nom de ce paquet (par exemple grâce à la section précédente), on l'écrit dans la case *Rechercher*, et on sélectionne *Nom* dans la liste déroulante nommée *Rechercher dans*.

page 116

16.3.4 Sélectionner le paquet à installer

Vient alors la phase d'installation proprement dite du paquet trouvé précédemment. Il y a différentes façons de le faire, selon que l'on souhaite utiliser la version disponible dans les dépôts officiels de sa distribution, ou un paquet provenant d'un autre dépôt, par exemple pour avoir une version plus récente.

Pour installer la version par défaut

Normalement, le paquet désiré se trouve maintenant quelque part dans la liste de paquets. Une fois trouvée la ligne correspondante, on clic-droit dessus, et dans le menu contextuel on choisit *Sélectionner pour installation*.

Si ce paquet dépend d'autres paquets, le gestionnaire de paquets ouvre alors une fenêtre où il demande s'il doit *Prévoir d'effectuer d'autres changements ?* En général, ses propositions sont pertinentes, et on peut accepter en cliquant sur *Ajouter à la sélection*.

Pour installer une version particulière

Parfois, on souhaite installer une version particulière d'un paquet parmi celles disponibles. Par exemple, si on a ajouté des dépôts spécifiques. Au lieu de choisir *Sélectionner pour installation* dans le menu contextuel, il faut sélectionner le paquet désiré d'un clic gauche, puis choisir, dans le menu *Paquet*, de *Forcer la version…* La suite ne change pas.

page 122

16.3.5 Appliquer les modifications

Il est possible de répéter les deux dernières étapes pour installer plusieurs paquets en même temps. Une fois qu'on a préparé cette installation, il ne reste qu'à la lancer en cliquant sur *Appliquer* dans la barre d'outils. Le gestionnaire de paquets ouvre alors une fenêtre *Résumé* où il liste tout ce qu'il va faire. Après avoir jeté un œil pour vérifier qu'on ne s'est pas trompé, on clique sur *Appliquer*.

Le gestionnaire de paquets télécharge alors les paquets depuis Internet, les vérifie, puis les installe. Il peut arriver que le gestionnaire indique que certains paquets n'ont pas pu être vérifiés : cette information n'est pas à prendre à la légère. Dans un tel cas, il vaut mieux annuler le téléchargement, cliquer sur *Recharger* dans le menu principal, et recommencer l'opération de sélection des paquets. Si l'indication apparaît de nouveau, cela peut être le fruit d'une attaque, d'une défaillance technique ou de soucis de configuration. Mais autant s'abstenir d'installer de nouveaux paquets avant d'avoir identifié la source du problème.

Enfin, si tout s'est bien passé, le gestionnaire de paquets affiche une fenêtre comme quoi *Les modifications ont été appliquées* et on peut donc cliquer sur *Fermer*. C'est alors une bonne idée de fermer le gestionnaire de paquets, pour éviter qu'il tombe entre d'autres mains.

16.4 Comment modifier ses dépôts Debian

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : un quart d'heure à une demi-heure.

Les paquets Debian qui contiennent les programmes se trouvent dans ce qu'on appelle des *dépôts*. Si les dépôts fournis avec Debian contiennent quasiment tous les logiciels dont on peut avoir besoin, il est parfois utile d'installer de nouveaux dépôts, comme *backports.debian.org* qui contient des programmes plus récents que ceux inclus dans la distribution stable de Debian, ou *deb-multimedia.org* qui contient des *codecs* et des logiciels multimédia non-libres, ou interdits dans certains pays, par exemple pour des questions de brevets.

Attention : ajouter un nouveau dépôt Debian sur un ordinateur revient à décider de faire confiance aux gens qui s'en occupent. Si les dépôts de *backports.debian.org* sont maintenus par des membres de Debian, ce n'est pas le cas pour de nombreux autres dépôts. La décision de leur faire confiance ne doit pas se prendre à la légère : si le dépôt en question contient des logiciels malveillants, il serait possible de les installer sur l'ordinateur sans même s'en rendre compte.

page 24

16.4.1 Authenticité du contenu des dépôts

Les dépôts Debian sont signés par des clés GnuPG. Ceci afin de s'assurer que leur contenu n'a pas été altéré par malveillance ou simple problème technique.

Cette section traite rapidement de comment chercher et vérifier une clé GnuPG. Il s'agit de cryptographie asymétrique, technique qui sera traitée plus avant dans le tome 2. On se contentera ici de donner un protocole permettant de vérifier une clé à partir d'une empreinte (ou « *fingerprint* »). On **n'**apprendra **pas** à bien utiliser la cryptographie asymétrique.

tome 2 ch. 6

Ce protocole simplifié a des limites : en particulier, on vérifie la clé à partir d'une empreinte (une sorte de somme de contrôle). Toute la confiance que nous allons donner à la clé vient uniquement de cette empreinte. Or vérifier cette empreinte à partir de ce qui est écrit dans ce guide, cela signifie faire fortement confiance à la source à partir de laquelle on l'a obtenu. En lisant ce guide sur Internet, c'est encore pire : on fait, en plus, confiance à sa connexion Internet.

page 41

Encore une fois, tout est affaire de compromis entre utilisabilité et sécurité. Pour obtenir des empreintes en toute confiance, le mieux est de les vérifier en tête-à-tête. Malheureusement, ce n'est généralement pas possible en pratique lorsqu'il s'agit de dépôts Debian. Ce n'est cependant pas une raison pour ne rien vérifier du tout.

Dans le cadre du premier tome de ce guide, qui n'aborde pas les problématiques liées à l'utilisation de réseaux, il n'y a pas de solution vraiment satisfaisante. En attendant le tome 2, qui traitera plus avant de ces questions, le mieux qu'on ait trouvé est d'utiliser :

- des empreintes données dans ce guide, qui ont été vérifiées sur Internet à partir de nombreuses connexions différentes à plusieurs moments différents, mais impliquent de faire confiance à la source de ce guide ;
- si possible, les empreintes qui se trouvent sur d'autres ordinateurs sur lesquels les dépôts en question auraient été installés précédemment si l'on peut y avoir accès, chez des proches par exemple.

Ce protocole est vraiment loin d'être sûr. Il met cependant des bâtons dans les roues de l'éventuelle personne qui souhaiterait nous faire installer des logiciels malveillants.

Quelques empreintes vérifiées par nos soins

Deux des empreintes de dépôts parmi les plus utilisés sont reproduites ci-dessous :

Dépôt	Date	Empreinte
`deb-multimedia.org`	octobre 1999	`1D7F C53F 80F8 52C1 88F4 ED0B 07DC 563D 1F41 B907`
`deb.torproject.org`	septembre 2009	`A3C4 F0F9 79CA A22C DBA8 F512 EE8C BC9E 886D DD89`

Comparer les empreintes avec celles présentes sur d'autres ordinateurs

Si on peut avoir accès à des ordinateurs sur lesquels les dépôts que l'on souhaite utiliser ont déjà été installés, on pourra recouper les empreintes données dans ce guide avec celles présentes sur ces ordinateurs.

Pour ce faire, sur des ordinateurs variés, ouvrir un *Terminal administrateur* à partir du menu *Applications* → *Accessoires*.

Taper alors :

```
apt-key finger
```

Puis appuyer sur *Entrée*. On obtient alors une liste des clés de dépôts, chacune sous la forme suivante :

```
pub   2048R/886DDD89 2009-09-04 [expire :2014-09-03]
Empreinte de la clé = A3C4 F0F9 79CA A22C DBA8 F512 EE8C BC9E 886D DD89
uid                  deb.torproject.org archive signing key
sub   2048R/219EC810 2009-09-04 [expire :2012-09-03]
```

C'est la troisième ligne de chaque entrée qui donne le nom du dépôt. Il s'agit dans cette liste de trouver le nom du dépôt qui nous nous intéresse. Dans l'exemple ci-dessus, on a :

```
uid                  deb.torproject.org archive signing key
```

Il s'agit donc de la clé de *deb.torproject.org*. L'empreinte correspondante se trouve sur la ligne juste au-dessus :

```
Empreinte de la clé = A3C4 F0F9 79CA A22C DBA8 F512 EE8C BC9E 886D DD89
```

Noter alors cette empreinte pour de futures comparaisons.

Récupérer la clé d'un dépôt depuis Internet

Il faut tout d'abord ouvrir *Applications* → *Outils système* → *Préférences* → *Mots de passe et clés*.

- Dans le menu *Distant* choisir *Chercher des clés distantes* ;
- Dans *Chercher des clés contenant*, taper une partie du nom de la clé recherchée ou son identifiant, par exemple « torproject.org » ou « 1F41B907 » (pour deb-multimedia), puis cliquer sur *Chercher* ;

- Une fenêtre *Clés distantes contenant […]* s'ouvre. Ici on a par exemple « deb.torproject.org archive signing key » qui a pour identifiant `886DDD89` ;
- Cliquer sur cette clé, puis sur le bouton *Importer*.
- On peut alors fermer la fenêtre avec la liste des clés trouvées à distance.

Afin de s'assurer que la clé que l'on vient d'obtenir est bien celle qu'on attend, il s'agit maintenant de vérifier son empreinte :

- Une fois la clé importée, aller dans l'onglet *Affichage* de la fenêtre principale.
- sélectionner l'affichage *Par trousseau*, puis *Tout afficher*
- Dans le nouveau menu, à gauche, sélectionner *Clés GnuPG*
- Sélectionner la clé à vérifier, dans notre exemple « deb.torproject.org archive signing key ».
- Cliquer dessus avec le bouton droit de la souris, et dans le menu contextuel qui apparaît, choisir *Propriétés*.
- Aller dans l'onglet *Détails*.
- Vérifier que l'empreinte correspond à celle relevée par ailleurs pour s'assurer qu'on a la bonne clé.

Si c'est bien le cas, on peut exporter la clé dans un fichier, avant de l'ajouter au logiciel qui s'en servira pour vérifier le contenu des dépôts. Pour cela, fermer la fenêtre avec les propriétés, sélectionner la clé, puis dans *Fichier* cliquer sur *Exporter…* puis l'enregistrer, par exemple dans *Documents*, en acceptant le nom par défaut. On peut fermer *Mots de passe et clés de chiffrement*.

16.4.2 Ajouter un nouveau dépôt

À partir du menu *Applications* → *Outils système* → *Administration* → *Gestionnaire de paquets Synaptic*. Puisque ce logiciel permet de choisir à quels programmes on fait confiance, on est rassuré qu'il nous demande notre mot de passe pour s'ouvrir. Aller dans le menu *Configuration*, puis *Dépôts*.

Configurer l'emplacement du dépôt

Aller dans l'onglet *Other Software*, ce qui signifie « Autre logiciel » (qui ne sont pas founis officiellement par le projet Debian) et cliquer sur le bouton *Add…* (Ajouter).

Entrer l'adresse du dépôt à ajouter dans la case *Ligne APT* de la boîte de dialogue qui s'ouvre. Par exemple, pour ajouter les *backports* (programmes plus récents que ceux inclus dans la distribution Debian stable) il faut entrer :

```
deb http ://ftp.fr.debian.org/debian wheezy-backports main
```

Si l'on souhaite également installer des logiciels non-libres, on peut ajouter `contrib` et `non-free` en plus de `main`. À la place de la ligne précédente, on pourrait plutôt entrer :

```
deb http ://ftp.fr.debian.org/debian wheezy-backports main contrib non-free
```

La plupart des dépôts que vous voudrez rajouter ici ne dépendent pas du tout du Projet Debian, par exemple, pour rajouter les dépôts du Projet TOR, il faudrait entrer :

```
deb http ://deb.torproject.org/torproject.org wheezy main
```

Une fois que c'est fait, il suffit de cliquer sur *Ajouter une source de mise à jour*.

Pour être sûr que le nouveau dépôt et les paquets qu'il contient sont bien ceux que l'on croit, on doit disposer d'un fichier qui contient la clé avec laquelle sont signées les listes de paquets du dépôt à ajouter, et avoir vérifié cette clé. La télécharger sur un site web et lui faire confiance aveuglément n'est pas une bonne idée.

tome 2 § 18.2

Ajouter une nouvelle clé de confiance

Le dépôt contenant les *backports* est devenu un dépôt officiel de Debian avec la sortie de la version *Squeeze* de Debian. La clé de confiance est donc la même que pour le reste des paquets officiels.

Par contre, si on souhaite utiliser le dépôt `deb.torproject.org`, fourni par le projet Tor, il est nécessaire d'ajouter sa clé de confiance. Après suivi la procédure de vérification de la clé, et avoir exporté cette dernière, on procède ainsi :

- Aller dans l'onglet *Authentification* et cliquer sur le bouton *Import Key File…* (importer un fichier de clé) ;
- Sélectionner le fichier où on a précédemment sauvegardé la clé téléchargée — `deb.torproject.org archive signing key.pgp` dans le dossier *Documents* si l'on a suivi les conseils précédents — et cliquer sur valider. On pourra ensuite supprimer le fichier en question.

Mettre à jour les paquets disponibles

Il est maintenant possible de refermer les *Sources de mises à jour*. De retour dans le *Gestionnaire de paquets Synaptic*, il faut cliquer sur *Recharger* pour mettre à jour les listes de paquets.

Installer le paquet avec les clés du dépôt

Une fois la clé ajoutée, on a accès au dépôt. Celui-ci fournit généralement un paquet contenant les clés de ce dépôt, et permettant de le mettre aisément à jour. Il est souvent nommé à partir du nom du dépôt, suivi du mot `keyring`. Par exemple, pour *deb.torproject.org*, il s'agit de `deb.torproject.org-keyring`. Il faut donc prendre le temps d'installer ce paquet, s'il est disponible.

page 120

Chapitre 17

Effacer des données « pour de vrai »

On a vu dans la première partie que lorsqu'on efface un fichier, son contenu n'est pas vraiment supprimé. Cependant, il existe des programmes qui permettent d'effacer des fichiers *et leur contenu*, ou du moins qui tentent de le faire, avec les limites expliquées auparavant.

page 32
page 32

17.1 Un peu de théorie

Pour la plupart des prochaines recettes, nous allons utiliser la suite de logiciels *secure-delete*.

17.1.1 La méthode de Gutmann

La documentation [1] du paquet *secure-delete*, inspirée d'une publication de Peter Gutmann publiée en 1996 [2], nous dit (en anglais) :

Le processus d'effacement fonctionne comme suit :

1. *la procédure d'écrasement (en mode sécurisé) remplace le contenu du fichier à 38 reprises. Après chaque passage, le cache du disque est vidé ;*
2. *le fichier est tronqué, de sorte qu'un attaquant ne sache pas quels blocs du disque appartenaient au fichier ;*
3. *le fichier est renommé, de sorte qu'un attaquant ne puisse tirer aucune conclusion sur le contenu du fichier supprimé à partir de son nom ;*
4. *finalement, le fichier est supprimé. […]*

17.1.2 Le compromis adopté

L'étude de Peter Gutmann porte sur des technologies de disques durs qui n'existent plus de nos jours. Il a depuis ajouté, à la fin de son article, un paragraphe intitulé *Epilogue* qui nous dit, en substance, que pour un disque dur « récent » [3], les 38 écritures successives ne sont plus nécessaires : il suffit d'écraser les données quelques fois avec des données aléatoires. Mais mis à part la nature et le nombre de réécritures, le processus décrit précédemment reste tout à fait d'actualité.

De surcroît, le NIST (*National Institute of Standards and Techonology*, organisme gouvernemental états-unien définissant les protocoles de sécurité utilisés, entre autres, par les administrations de ce pays) a publié une étude récente [4] de la NSA, qui semble

1. Fichier `README.gz` installé sur une Debian dans `/usr/share/doc/secure-delete`.
2. Peter Gutmann, 1996, *Secure Deletion of Data from Magnetic and Solid-State Memory* [http://www.cs.auckland.ac.nz/~pgut001/pubs/secure_del.html] (en anglais).
3. Utilisant la technologie PRML [https://fr.wikipedia.org/wiki/PRML], apparue en 1990 [http://www.storagereview.com/guide/histFirsts.html].
4. NIST, 2006, *Guidelines for Media Sanitization* [http://csrc.nist.gov/publications/nistpubs/800-88/NISTSP800-88_with-errata.pdf] (en anglais).

conclure que sur les disques durs modernes, les données sont tellement collées les unes aux autres qu'il devient impossible de se livrer à des analyses magnétiques pour retrouver les traces de données effacées ; en effet, la densité des données des disques durs ne cesse de croître, afin d'augmenter leur capacité de stockage.

Par conséquent, nous nous contenterons dans les recettes qui suivent de quelques passages aléatoires, en évoquant tout de même la mise en œuvre de la méthode originale de Gutmann.

page 51 Il s'agira une fois de plus de faire le bon compromis, au cas par cas, entre la rapidité et le niveau de protection souhaité, en fonction de la taille des données à écraser, de l'âge du disque dur, et de la confiance qu'on accorde au NIST.

17.1.3 Pour les clés USB et autres mémoires *flash*

Pour les clés USB (ou autre mémoire *flash*), comme les cartes SD, ou disques durs SSD, une étude datant de 2011 [5] a montré que la situation était réellement problématique.

Cette étude démontre qu'il est impossible d'avoir la garantie d'avoir recouvert tout le contenu d'un fichier donné, quel que soit le nombre de réécritures. Même si cela rend inaccessibles les données en branchant simplement la clé, elles sont toujours visibles pour quiconque regarderait directement dans les puces de mémoire *flash*.

La seule méthode qui a fonctionné de façon systématique était de réécrire plusieurs fois l'*intégralité* de la clé USB. Dans la plupart des cas, deux passages ont suffit, mais sur certains modèles, vingt réécritures ont été nécessaires avant que les données ne disparaissent pour de bon.

page 137 Partant de ces constats, la réponse préventive semble être de chiffrer systématiquement les clés USB, opération rendant vraiment plus difficile d'extraire des informations directement depuis les puces de mémoire *flash*. Et pour nettoyer a posteriori, l'écrasement entier, malgré ses limites, protège tout de même contre les attaques purement logicielles.

17.1.4 D'autres limites de l'effacement « sécurisé »

Il peut encore rester des informations sur le fichier permettant de le retrouver, notamment si l'on utilise un système de fichiers journalisé comme *ext3*, *ext4*, ReiserFS, XFS, JFS, NTFS, un système d'écriture, de compression ou de sauvegarde, sur disque (exemple : RAID) ou via un réseau. Voir à ce sujet la page 33 première partie.

17.2 Sur d'autres systèmes

page 29 On a vu qu'il est illusoire, si l'on utilise un système d'exploitation propriétaire, de rechercher une réelle intimité. Bien qu'il existe des logiciels supposés effacer des fichiers avec leur contenu sous Windows et Mac OS X, il est donc bien plus difficile de leur faire confiance.

17.3 Allons-y

On peut effacer le contenu :

- de fichiers individuels, voir page ci-contre ;
- de tout un périphérique, voir page suivante ;
- de fichiers déjà supprimés, voir page 135.

5. Michael Wei et Al, 2011, *Reliably Erasing Data From Flash-Based Solid State Drives* [https://www.usenix.org/legacy/events/fast11/tech/full_papers/Wei.pdf] (en anglais).

17.4 Supprimer des fichiers... et leur contenu

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 5 minutes de préparation, puis quelques secondes à plusieurs heures d'attente en fonction de la taille du fichier à effacer et de la méthode utilisée.

Voici donc la méthode à suivre pour se débarrasser de fichiers, en prenant soin de rendre illisible ce qu'ils contenaient.

Attention ! Après avoir recouvert le contenu de fichiers sur une clé USB (ou tout autre support de stockage utilisant de la mémoire flash comme une carte SD ou disque dur SSD) il y a de fortes chances qu'il se trouve encore inscrit dans une région inaccessible de la clé !

17.4.1 Installer les logiciels nécessaires

Si ce n'est pas déjà fait, il nous faut installer `nautilus-wipe`. Ce paquet est présent par défaut dans *Tails*. page 120

17.4.2 Supprimer des fichiers et leur contenu à partir du navigateur de fichiers

Dans *Tails*

Afin de supprimer des fichiers et leur contenu en utilisant *Tails*, consulter la documentation en cliquant sur l'icône *Documentation de Tails* se trouvant sur le bureau. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Chiffrement et vie privée* et cliquer sur la page *Effacer des fichiers de façon sécurisée et nettoyer l'espace disque avec Nautilus Wipe*.

Avec une Debian chiffrée

Afin de supprimer des fichiers et leur contenu depuis le navigateur de fichiers, naviguer jusqu'au fichier, faire un clic-droit sur celui-ci et sélectionner *Écraser*. Une fenêtre s'ouvre nous demandant de confirmer la suppression, et proposant également quelques *Options*.

Nous pouvons choisir le nombre de passes effectuées afin de recouvrir les données de notre périphérique ainsi que quelques options de comportement lors de l'effacement des données. Les options par défaut sont suffisantes pour les disques durs actuels.

Cliquer ensuite sur *Écraser*. Une fois l'effacement terminée, une fenêtre *L'écrasement a réussi* s'ouvre, précisant que *Le(s) élément(s) ont été écrasé(s) avec succès.*.

17.5 Effacer « pour de vrai » tout un disque

Avant de se débarrasser d'un disque dur, de le recycler, de réinstaller un système propre, ou encore d'envoyer un ordinateur en panne au Service Après Vente, il peut être judicieux de mettre des bâtons dans les roues des gens qui voudraient récupérer les données qu'il contenait. Pour cela, la meilleure solution est encore de les remplacer par du charabia. page 57

Avant d'utiliser cette recette, il faut réfléchir à deux fois et sauvegarder soigneusement les données à conserver. Si elle est bien appliquée, elle rend en effet les données très difficiles à récupérer, même en analysant le disque dans un laboratoire. page 143

Nous verrons d'abord comment effacer tout le contenu d'un disque, puis comment rendre le contenu d'une partition chiffrée inaccessible rapidement.

17.6 Effacer tout le contenu d'un disque

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 5 minutes de préparation, puis plusieurs heures d'attente en fonction de la taille du disque.

Pour effacer un volume complet (disque ou partition), on va utiliser la commande `shred` de façon à ce qu'elle recouvre la totalité des données trois fois avec des données aléatoires. Cette commande permet donc, en plus de l'effacement des fichiers, de recouvrir l'espace effacé de telle manière qu'il devient quasiment impossible de retrouver ce qu'il contenait auparavant.

Pour recouvrir le contenu d'un disque, il est nécessaire de ne pas être en train de l'utiliser… s'il contient le système d'exploitation habituellement utilisé, il faut donc mettre le disque dur dans un autre ordinateur ou utiliser un système *live*. `shred` étant un outil standard, n'importe quel système *live* devrait faire l'affaire.

page 99

La commande est très simple. Elle exige seulement de connaître l'emplacement du périphérique (son chemin) que l'on veut effacer, puis de faire preuve de patience car le processus prend plusieurs heures.

17.6.1 Trouver le chemin du périphérique

Avant tout, il faut savoir repérer sans se tromper le chemin utilisé par le système d'exploitation pour désigner le support de stockage qu'on veut effacer.

Si l'on souhaite effacer un disque interne, commencer par débrancher tous les disques durs externes, clés USB, lecteurs de cartes mémoire ou autre périphérique de stockage branchés sur l'ordinateur. D'une part, cela évitera de les effacer par erreur ; d'autre part, cela rendra la recherche du disque interne plus facile.

Bien sûr, il ne faut pas faire cela si c'est justement le contenu d'un disque externe que l'on souhaite rendre inaccessible.

Ouvrir l'Utilitaire de disque

Pour trouver le chemin du périphérique, une méthode simple consiste à utiliser l'*Utilitaire de disque* que l'on démarre par le menu *Applications* → *Accessoires*.

Chercher le chemin du périphérique

La partie située à gauche indique la liste des disques durs connus du système. On peut cliquer sur l'un d'entre eux afin de voir plus d'informations apparaître sur la partie droite. Les icônes, la taille indiquée ainsi que le nom des disques devraient permettre d'identifier celui que l'on cherche.

Si cela ne suffit pas, il est possible de jeter un œil à l'organisation des partitions, en regardant le tableau qui apparaît dans la partie droite :

- si le disque à effacer contenait un système GNU/Linux non chiffré, il doit y avoir au moins deux partitions, l'une avec un système de fichiers *swap*, l'autre en général *ext3* ;
- si le disque à effacer contenait un système GNU/Linux chiffré, il doit y avoir au moins deux partitions, l'une avec un système de fichiers *ext2*, l'autre en général *Chiffré* ou *Inconnu* ;
- si le disque à effacer contenait un système Windows, il doit y avoir une ou plusieurs partitions notées *ntfs* ou *fat32*.

Par ailleurs, le périphérique correspondant au disque interne est généralement le premier de la liste.

Une fois le disque trouvé et sélectionné, on pourra lire le chemin du disque dans la partie à droite, sous l'intitulé *Disque*, à côté de l'étiquette *Périphérique*.

Le chemin du périphérique commence par `/dev/` suivi de trois lettres, les deux premières étant `sd` ou `hd` : par exemple, `/dev/sdx`. Noter le chemin quelque part : il faudra l'écrire tout à l'heure à la place de `LE_PÉRIPHÉRIQUE`.

Attention ! Ce chemin n'est pas nécessairement toujours le même. Il vaut mieux recommencer cette courte procédure après avoir redémarré l'ordinateur, branché ou débranché une clé USB ou un disque dur. Cela évitera les mauvaises surprises… comme perdre le contenu d'un *autre* disque dur.

17.6.2 Lancer la commande *shred*

Ouvrir un *Terminal administrateur* à partir du menu *Applications* → *Accessoires*. page 85

Saisir la commande suivante en veillant à remplacer `LE_PÉRIPHÉRIQUE` par le chemin de périphérique déterminé précédemment :

```
shred -n 3 -v LE_PÉRIPHÉRIQUE
```

Si l'on préfère utiliser la méthode originale de Gutmann (plus longue, et peut-être plus sûre), il faut remplacer `-n 3` par `-n 25` dans la ligne de commande.

Une fois la commande tapée et vérifiée, appuyer sur la touche *Entrée*. La commande `shred` va alors écrire dans le terminal ce qu'elle fait (ainsi qu'on lui a demandé de le faire en ajoutant à la commande `shred` l'option `-v`, qui signifie, dans le cadre de *cette* commande, que l'ordinateur doit être « verbeux » — c'est-à-dire « bavard ») :

```
shred :/dev/sdb :pass 1/3 (random)...
shred :/dev/sdb :pass 2/3 (random)...
shred :/dev/sdb :pass 3/3 (random)...
```

À la fin de la procédure, le terminal affiche à nouveau le signe `#`, qui symbolise l'invite de commande. On peut alors fermer le terminal.

17.6.3 Réutiliser le disque

Attention, cette méthode efface non seulement les données d'un volume complet mais, à la fin de l'opération, le disque n'a plus ni table de partitions, ni système de fichiers. Pour le réutiliser, il est nécessaire de créer entièrement au moins une nouvelle partition et son système de fichiers, avec l'*Utilitaire de disque* par exemple. page 16 page 16

17.7 Effacer le contenu d'une partition chiffrée LUKS

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 10 minutes environ.

Il est extrêment rapide de rendre inaccessible le contenu d'une partition LUKS, le format de stockage standard des clés de disques chiffrés sous GNU/Linux.

page 37

Certains logiciels de chiffrement d'un disque complet ont en effet la capacité de détruire la clé de chiffrement, rendant ainsi le contenu chiffré incompréhensible. Vu que la clé contient une part minuscule d'informations et peut être détruite presque instantanément, cette méthode est une alternative bien plus rapide à l'écrasement de l'ensemble des données.

Ceci dit, cette option n'est réalisable que si le disque dur a déjà été chiffré. Si les données confidentielles contenues sur le disque ne sont pas déjà chiffrées, il est nécessaire d'effacer le disque entier, comme expliqué précédemment, avant de pouvoir s'en débarrasser ou le réutiliser en toute tranquillité.

page 130

Attention ! Sur une clé USB (ou tout autre support de stockage utilisant de la mémoire *flash*), cette méthode ne permet pas de garantir que les données ainsi recouvertes soient réellement rendues inaccessibles. Il est donc nettement plus sûr de recouvrir plusieurs fois l'intégralité des données de la clé.

page 130

17.7.1 Repérer la partition en question

Comme dans le cas précédent, si l'on souhaite effacer un disque interne, commencer par débrancher tous les disques durs externes, clés USB, lecteurs de cartes mémoire ou autre périphérique de stockage branchés sur l'ordinateur. D'une part, cela évitera de les effacer par erreur ; d'autre part, cela rendra la recherche du disque interne plus facile.

Bien sûr, il ne faut pas faire cela si c'est justement le contenu d'un disque externe que l'on souhaite rendre inaccessible.

Ouvrir l'Utilitaire de disque

On va s'aider de l'*Utilitaire de disque*, que l'on ouvre à partir du menu *Applications* → *Accessoires*.

Chercher le disque en question

La partie située à gauche indique la liste des disques durs connus du système. Les icônes, la taille indiquée ainsi que le nom des disques devraient permettre d'identifier celui que l'on cherche. On peut cliquer sur l'un d'entre eux afin de voir plus d'informations apparaître sur la partie droite.

Le premier périphérique que l'*Utilitaire de disque* affiche dans cette liste est généralement le disque dur interne.

Chercher le chemin de la partition

Lorsqu'un disque est sélectionné dans la liste de gauche, plus d'informations à son sujet apparaissent dans la partie droite de la fenêtre. On s'intéressera en particulier au schéma des *Volumes*, ou partitions.

Normalement, les partitions chiffrées sont indiquées dans la partie *Volumes* avec comme indication *Chiffré…* mais ce n'est pas toujours le cas : l'*Utilitaire de disque*

peut aussi indiquer que le type de partition est *inconnu*, ou parfois indiquer un type plus courant (*ext3*, *NTFS*).

Une fois la partition chiffrée à effacer repérée sur le schéma, cliquer dessus. Son chemin apparaît alors sous le schéma à droite, à côté de l'étiquette *Périphérique*. C'est ce chemin que nous allons noter. Nous l'appellerons `LE_PÉRIPHÉRIQUE_CHIFFRÉ`. Ce doit être quelque chose comme `/dev/sdx9`.

17.7.2 Ouvrir un terminal administrateur

À partir du menu *Applications* → *Accessoires*, ouvrir un *Terminal administrateur*. page 85

17.7.3 Vérifier le premier repérage et récupérer la taille de l'en-tête LUKS

Dans le terminal, la commande `cryptsetup luksDump` donne plein d'informations sur l'en-tête LUKS, dont sa taille sur le disque (en secteurs de 512 octets). Taper donc, en remplacant `LE_PÉRIPHÉRIQUE_CHIFFRÉ` par la valeur déterminée ci-dessus :

```
cryptsetup luksDump LE_PÉRIPHÉRIQUE_CHIFFRÉ
```

Dans le cas où on se serait trompé de périphérique, le terminal ne renvoie soit aucune réponse, soit :

```
LE_PÉRIPHÉRIQUE_CHIFFRÉ n'est pas un périphérique LUKS valide.
```

Si l'on ne s'est pas trompé, on doit plutôt se voir répondre quelque chose comme :

```
LUKS header information for /dev/sdx2

Version :       1
Cipher name :   aes
Cipher mode :   cbc-essiv:sha256
Hash spec :     sha1
Payload offset :4096
MK bits :       256
MK digest :     a4 79 85 49 1f 3f 71 e5 1e c6 07 14 88 0c 02 27
                59 80 25 58
MK salt :       b7 b1 2a 5d 6d c5 b5 d2 06 55 a3 85 5d 07 af 9b
                c9 03 46 c6 e6 2f 29 1a 9d b7 58 05 44 cc 68 f9
MK iterations : 54520
UUID :          d73cbb8a-058f-469e-935a-7f71debd8193

Key Slot 0 : ENABLED
    Iterations :        217000
    Salt :              ec 1e 63 b7 13 fb 20 21 18 5d 86 44 42 d0
                        f2 af 52 a4 74 54 22 3f d8 0b ad 69 8c 46
                        f2 d3 79 4d
    Key material offset :8
    AF stripes :        4000
```

On va avoir besoin de la taille de l'en-tête (en secteurs), écrite sur la ligne `Payload offset` : la noter quelque part. On l'utilisera plus loin sous le nom d'`OFFSET`.

17.7.4 Recouvrir l'en-tête LUKS de données aléatoires

Comme dans la recette précédente, on va utiliser la commande `shred` pour écraser les données, mais cette fois on écrasera uniquement l'en-tête LUKS (cet en-tête contient la clé qui permet de déchiffrer le reste des données). Cela ira donc beaucoup plus vite.

Dans le terminal administrateur, saisir la commande suivante en prenant bien soin de remplacer `OFFSET` et `LE_PÉRIPHÉRIQUE_CHIFFRÉ` par les valeurs qu'on a trouvées :

```
shred -n 3 -s $((OFFSET * 512)) -v LE_PÉRIPHÉRIQUE_CHIFFRÉ
```

Appuyer alors sur *Entrée*.

page 85

Note : l'option `-s` utilisée ici sert, dans le cadre de *cette* commande, à indiquer la taille (*size*) de l'espace qui doit être effacé de manière sécurisée.

Une fois que le terminal rend la main, les données chiffrées devraient être illisibles. Pour s'en assurer, il est possible de chercher un en-tête LUKS qui n'aurait pas été bien effacé en tapant à nouveau :

```
cryptsetup luksDump LE_PÉRIPHÉRIQUE_CHIFFRÉ
```

Si l'en-tête a bien été effacé, le terminal renvoie soit aucune réponse, soit :

```
LE_PÉRIPHÉRIQUE_CHIFFRÉ n'est pas un périphérique LUKS valide.
```

page 130

Enfin il est possible, voire même conseillé, d'effacer quand même l'ensemble de la partition, en suivant la recette précédente.

17.8 Rendre irrécupérables des données déjà supprimées

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 5 minutes de préparation, puis de plusieurs minutes à plusieurs heures d'attente, selon la taille du disque à nettoyer et selon la méthode utilisée.

Lorsque des fichiers ont *déjà* été effacés sans précautions particulières, les données qu'ils contenaient se trouvent toujours sur le disque. L'objectif de cette recette est de recouvrir les données qui subsisteraient, en écrasant l'espace libre d'un disque dur.

Attention ! Comme les autres façons d'effacer un fichier « pour de vrai », cela ne marche pas avec certains systèmes de fichiers « intelligents » qui, pour être plus efficaces, ne vont pas montrer tout l'espace libre au logiciel chargé d'y recouvrir les traces. Voir à ce sujet la première partie. Il ne faut pas non plus faire confiance à cette méthode pour une clé USB, les cartes SD ou disques durs SSD et préférer recouvrir plusieurs fois l'intégralité des données qu'elle contient.

page 33 page 130

Dans *Tails*

Le paquet `nautilus-wipe` est déjà installé par défaut dans Tails. Il nous suffit donc de consulter la documentation, en cliquant sur l'icône *Documentation de Tails* se trouvant sur bureau.

Dans le menu à droite, cliquer sur *Documentation*. Puis, dans l'index qui s'ouvre, chercher la section *Chiffrement et vie privée* et cliquer sur la page *Effacer des fichiers de façon sécurisée et nettoyer l'espace disque avec Nautilus Wipe*.

Avec une Debian chiffrée

Si ce n'est pas déjà fait, il nous faut installer le paquet, voir page 120 `nautilus-wipe`.

Il nous faut ensuite ouvrir un navigateur de fichiers, puis naviguer jusqu'au disque que l'on veut nettoyer. Effectuez ensuite un clic-droit dans la partie droite du navigateur de fichiers et sélectionnez *Écraser l'espace disque disponible*. Une fenêtre s'ouvre nous demandant de confirmer la suppression de l'espace disque disponible, et proposant également quelques *Options*.

Nous pouvons choisir le nombre de passes effectuées afin de recouvrir les données de notre périphérique, ainsi que quelques options de comportement lors de l'effacement des données. Les options par défaut sont suffisantes pour les disques durs actuels.

Cliquer ensuite sur *Écraser l'espace disque disponible*. L'effacement peut prendre du temps. Remarquons qu'un fichier appelé ooooooooo.ooo est créé dans le dossier. Nautilus Wipe va en augmenter la taille autant que possible, afin d'utiliser tout l'espace libre disponible, puis l'écrasera de manière sécurisée. Une fois l'effacement terminé, une fenêtre *L'écrasement a réussi* s'ouvre, précisant que *L'espace disque disponible sur la partition ou le périphérique « … » a été écrasé avec succès.*

Chapitre 18

Partitionner et chiffrer un disque dur

Nous allons maintenant aborder le chiffrement d'un périphérique, afin d'y stocker des données de manière chiffrée.

Une fois un disque dur chiffré, les données qu'il contient ne sont accessibles que lorsqu'on a tapé une phrase de passe permettant de le déchiffrer. Pour plus d'informations là-dessus, consultez la partie sur la cryptographie. page 37

Une fois la phrase de passe saisie, le système a accès aux données du disque dur en question, il ne faut donc pas taper cette phrase de passe n'importe où, mais seulement sur les ordinateurs et les systèmes dans lesquels on a suffisamment confiance. page 51

En effet, non seulement ceux-ci auront accès aux données déchiffrées, mais des traces de la présence du disque dur seront également gardées sur l'ordinateur. C'est pourquoi nous vous conseillons de l'utiliser sur un système GNU/Linux chiffré ou un système *live* amnésique. page 19 page 105 page 99

Il peut s'agir d'un disque dur externe, d'une clé USB, d'une carte SD ou encore d'une partie seulement d'un de ces périphériques. On peut en effet découper un disque dur ou une clé USB en plusieurs morceaux indépendants, qu'on appelle des partitions. page 16

Ci-dessous, on parlera de disque dur, sachant que, sauf mention contraire, le terme vaut aussi bien pour un disque dur externe, une clé USB, ou une carte SD.

Si on veut avoir un endroit sur le disque dur où mettre des données qui ne seront pas confidentielles, mais auxquelles on pourra accéder sur des ordinateurs non dignes de confiance, il est possible de découper le disque dur en deux partitions :

- une partition non chiffrée, où l'on ne met que des données non confidentielles, comme de la musique, que l'on peut utiliser depuis tous les ordinateurs sans taper la phrase de passe ;
- une partition chiffrée, avec les données confidentielles, qu'on n'ouvre que sur les ordinateurs auxquels on fait confiance.

18.1 Chiffrer un disque dur avec LUKS et `dm-crypt`

On va expliquer comment chiffrer un disque avec les méthodes standards sous GNU/Linux, appelées `dm-crypt` et LUKS. Ce système est maintenant bien intégré avec les environnements de bureau, et la plupart des opérations sont donc possibles sans avoir besoin d'outils particuliers.

18.2 D'autres logiciels que l'on déconseille

Il existe d'autres logiciels de chiffrement comme FileVault [1], qui est intégré dans Mac OS X — mais il s'agit d'un logiciel propriétaire — ou TrueCrypt — mais on a moins de raisons de lui faire confiance que le chiffrement standard de GNU/Linux, car il n'est plus mis à jour depuis mai 2014 et ce n'était pas vraiment un logiciel libre [2]. De plus, si l'on utilise un logiciel, même libre, sur un système d'exploitation propriétaire, on fait implicitement confiance à ce dernier car il a forcément accès aux données déchiffrées.

page 29
page 15

18.3 En pratique

Si le disque dur a déjà servi, il peut être nécessaire de commencer par recouvrir ses données, voir page 127.

Si le disque à chiffrer ne dispose pas d'espace libre, le formater, voir page ci-contre.

Ensuite, si l'on souhaite chiffrer une partie seulement du disque dur, il faut créer une partition en clair, voir page 140.

À la suite de quoi il ne reste plus qu'à l'initialiser pour contenir des données chiffrées, voir page 140.

Le voilà enfin prêt à être utilisé, voir page 141.

1. La dernière analyse indépendante de FileVault [http://crypto.nsa.org/vilefault/23C3-VileFault.pdf] date de 2006. En plus d'être sensible aux mêmes attaques que d'autres systèmes, FileVault a quelques faiblesses qu'il faut préciser : ce système ne permettant de chiffrer que le répertoire personnel, des traces seront écrites [page 19] en clair sur le reste du disque dur ; la phrase de passe de chiffrement est identique au mot de passe de la session, généralement faible ; le fait d'enregistrer un « mot de passe principal » ouvre un nouveau champ d'attaques ; la clé de chiffrement sera écrite sur le disque dur si *Utiliser la mémoire virtuelle sécurisée* n'a pas été choisi, ou si un ordinateur mis en veille vide sa batterie. Néanmoins, en gardant en tête que cela offre un niveau de protection limité, cela vaut tout de même la peine d'activer FileVault sur un ordinateur avec Mac OS X.

2. TrueCrypt, dont le développement a été interrompu [http://truecrypt.sourceforge.net/] en mai 2014, était distribué sous une licence particulière, la « *TrueCrypt License* » : le développement n'était pas ouvert, et seules les sources de la dernière version sont disponibles, ce qui rend plus difficile la vérification des modifications apportées. De plus, le logiciel n'est pas considéré comme libre par nombre de distributions GNU/Linux, notamment Debian, et ne correspond pas à la définition de l'*open source* de l'Open Source Initiative [http://www.opensource.org/docs/osd].

18.4 Préparer un disque à chiffrer

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : environ 10 minutes.

Ci-dessous, on parlera toujours de disque dur, sachant que ça vaut aussi bien pour un disque dur externe que pour une clé USB, sauf si on précise le contraire.

La procédure que l'on explique ici implique d'effacer toutes les données qui se trouvent sur le disque[3]. Si l'on a déjà de l'espace non partitionné sur son disque, on peut directement passer à l'étape de chiffrement.

page suivante

18.4.1 Installer les paquets nécessaires

Pour chiffrer notre disque dur, on a besoin d'avoir installé les paquets `secure-delete`, `dosfstools` et `cryptsetup`. Si l'on utilise Tails, ces paquets sont déjà installés.

page 120

18.4.2 Formater le disque dur avec l'Utilitaire de disque

Aller dans *Applications* → *Accessoires* → *Utilitaire de disque*.

Une fenêtre s'ouvre. La partie de gauche liste les disques connus du système. La partie de droite permet d'effectuer des actions.

Choisir le périphérique

À gauche, dans la section *Disques périphériques*, il y a la liste des disques. Si l'ordinateur utilisé contient un système chiffré, il y a aussi les volumes chiffrés de notre système.

Les icônes, la taille indiquée ainsi que le nom des disques devraient permettre d'identifier celui que l'on cherche.

Une fois le disque repéré, le sélectionner dans la liste.

Démonter les volumes

Si le volume est monté, un bouton *Démonter le volume* sera accessible dans le menu de droite, sous l'onglet *Volumes*. Cliquer sur ce bouton afin de démonter le volume. Si ce disque contient plusieurs volumes, les démonter tous, un par un.

Reformater le disque

Attention ! Formater un disque dur revient à supprimer tous les fichiers qui s'y trouvent.

Dans le menu de droite, sous l'onglet *Disque*, cliquer sur *Formater le disque*, puis choisir, comme *Schéma*, *Master Boot Record* et cliquer sur *Formater*. L'utilitaire de disque demande si l'on veut vraiment formater le périphérique. C'est le moment de vérifier que l'on a choisi le bon périphérique avant de faire une bêtise. Si c'est bien le cas, confirmer en cliquant sur *Formater*.

3. On pourrait également utiliser le logiciel GParted. Plus difficile à utiliser que l'*Utilitaire de disque*, ce dernier à l'avantage de savoir *redimensionner* une partition déjà existante tout en gardant les fichiers qui s'y trouvent.

18.5 Créer une partition non chiffrée

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 2 minutes.

On ouvrira la partition chiffrée qui contiendra nos données confidentielles uniquement sur les ordinateurs auxquels on fait confiance.

Si on le souhaite, c'est le moment de créer une partition non chiffrée où l'on met des données qui ne sont pas confidentielles, et que l'on peut utiliser depuis tous les ordinateurs sans avoir à taper la phrase de passe.

plus bas Si l'on désire chiffrer le disque dur en entier, on peut directement passer à l'étape suivante.

Toujours avec l'*Utilitaire de disque* et le disque dur sélectionné, dans la partie droite cliquer sur la zone *Libre* du schéma des *Volumes*. En-dessous, cliquer ensuite sur *Créer une partition*.

Placer le curseur *Taille* à la taille voulue pour la partition non chiffrée. L'espace laissé libre nous servira pour la partition chiffrée. Dans *Type*, choisir *FAT*. On peut aussi choisir un nom pour la partition. Une fois cela fait, cliquer sur *Créer*.

18.6 Créer une partition chiffrée

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 10 minutes + quelques minutes à plusieurs heures pour remplir l'espace libre, selon la taille de la partition.

18.6.1 Créer la partition chiffrée

Sur le schéma des *Volumes*, cliquer sur *Libre*. En-dessous, cliquer sur *Créer une partition* : une fenêtre s'ouvre.

Dans la section *Type*, choisir *FAT*. Entrer un nom pour la partition (en lettres, sans espaces ni caractères spéciaux, sinon ça risque fort de ne pas marcher) et cocher la case *Chiffrer le périphérique correspondant*. Valider en cliquant sur *Créer*.

page 91 Une fenêtre s'ouvre alors, qui demande la phrase de passe. Il faut en choisir une bonne, et la taper dans les deux cases correspondantes, avant de valider en cliquant sur *Créer*.

18.6.2 Remplir la partition de données aléatoires

Pour finir, on va remplir l'espace vide du disque dur de données aléatoires. Cela permet de cacher l'endroit où vont se trouver nos propres données, et complique donc la vie des personnes qui voudraient tenter de les déchiffrer.

Sur le schéma des *Volumes*, cliquer sur la partition *FAT* en-dessous de *Chiffré*, puis, à gauche sous le schéma, cliquer sur *Monter le volume*.

Ouvrir ensuite le dossier en le sélectionnant dans la liste située en-dessous de *Poste de travail* dans l'onglet *Raccourcis* du tableau de bord. Une fois le dossier correspondant à la clé ouverte on va suivre l'outil servant à rendre irrécupérables des données déjà

page 135

supprimées.

Le processus dure de quelques minutes à quelques heures, selon la taille du disque dur et sa vitesse (par exemple, 2 heures pour une clé USB de 4 Go).

18.6.3 Débrancher *proprement* le disque dur

Retourner dans l'*Utilitaire de disque*, cliquer sur *Démonter le volume*. Attendre un peu, puis sur le schéma des volumes, cliquer sur le volume *Chiffré* au-dessus du volume *FAT*. Cliquer alors sur *Verrouiller le volume* sous le schéma à droite (s'il y a un message d'erreur, ce n'est pas très grave), puis sur *Déconnexion en toute sécurité*, au-dessus du schéma à droite. Débrancher ensuite physiquement le disque externe de l'ordinateur. On peut désormais fermer l'*Utilitaire de disque*.

Le disque dur chiffré est maintenant utilisable.

18.7 Utiliser un disque dur chiffré

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 2 minutes, quelques heures… ou jamais, selon notre capacité à retenir la phrase de passe.

Afin de permettre au système d'accéder aux données qui se trouvent sur un disque chiffré, il est heureusement nécessaire d'indiquer la phrase de passe. Une opération plus ou moins simple selon les environnements…

18.7.1 Avec Debian (ou autre GNU/Linux)

Sur un système GNU/Linux avec un environnement de bureau configuré pour monter automatiquement les médias externes, une fenêtre apparaît pour demander la phrase de passe lorsqu'on branche un disque externe contenant des données chiffrées.

Si ce n'est pas le cas, elle apparaîtra quand on demandera au système de monter la partition, par exemple à partir du *Poste de travail*.

Pour fermer la partition chiffrée, il suffit de démonter le disque dur comme on le fait habituellement.

18.7.2 Avec d'autres systèmes

Nous ne connaissons pas de moyen simple d'accéder à la partition chiffrée du disque dur ni sous Windows, ni sous Mac OS X. Même si des solutions peuvent exister[4], il est bon de rappeler qu'il s'agit de systèmes d'exploitation propriétaires, en lesquels il n'y a aucune raison d'avoir confiance. page 29

Alors le mieux à faire, pour mettre sur le disque dur des données auxquelles on veut accéder sur des ordinateurs en lesquels on n'a pas confiance, c'est de mettre une deuxième partition, non chiffrée, sur le disque dur, comme expliqué précédemment. page précédente

4. Pour les anciennes versions de Windows (jusqu'à Vista), il était possible d'utiliser FreeOTFE (http://sourceforge.net/projects/freeotfe.mirror/).

Chapitre 19

Sauvegarder des données

Réaliser des sauvegardes est une opération relativement simple dans son principe : faire une copie des fichiers qu'on ne voudrait pas perdre, sur un autre support de stockage que celui où se trouvent les données.

Bien entendu, si on prend le soin de mettre nos données de travail sur des disques durs ou des clés USB chiffrés, il est nécessaire que ces copies soient chiffrées, elles aussi.

Deux autres points à ne pas négliger pour mettre en place une bonne *politique de sauvegardes* :

- définir une méthode pour effectuer **régulièrement** ses sauvegardes,
- tester de temps à autre si les sauvegardes sont toujours bien lisibles.

Ce dernier aspect est vraiment à ne pas négliger. Perdre les données originales est souvent pénible. S'apercevoir ensuite que les sauvegardes ne permettent pas de *restaurer* ce qu'on a perdu transforme la situation en catastrophe.

Dans le même ordre d'idée, cela parait également une bonne idée de ne pas stocker les sauvegardes au même endroit que les données originales. Sinon, on risque que les deux soient perdues ou détruites simultanément…

19.1 Gestionnaire de fichiers et stockage chiffré

Réaliser des sauvegardes est avant tout une question de rigueur et de discipline. Dans les cas simples, on peut se passer de logiciels spécialement prévus pour réaliser des sauvegardes, et se contenter simplement d'effectuer des copies avec le gestionnaire de fichiers.

19.1.1 Effectuer les sauvegardes

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : pour la première fois, le temps de chiffrer le support de stockage et de décider des fichiers à sauvegarder ; et ensuite cela dépend de la quantité de données à sauvegarder.

Le chiffrement de nos sauvegardes sera assuré par le chiffrement du support de stockage externe (clé USB ou disque dur). page 137

Pour effectuer les copies avec régularité et sans trop y passer de temps, il est recommandé :

- d'avoir quelque part une liste des fichiers et dossiers à sauvegarder ;
- de se fabriquer un petit calendrier des jours ou semaines où l'on fera ses sauvegardes, avec des cases que l'on cochera après les avoir faites.

Une bonne pratique consiste à créer un dossier avec la date de la sauvegarde pour y copier les données. Cela permet de garder facilement plusieurs sauvegardes si on le souhaite, et de supprimer tout aussi facilement les sauvegardes précédentes.

19.1.2 Restaurer une sauvegarde

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : selon la quantité de données à restaurer.

En cas de perte des données originales, la restauration se fait aussi simplement que la sauvegarde : en effectuant des copies dans l'autre sens.

19.1.3 S'assurer que les sauvegardes sont toujours lisibles

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : environ 5 minutes + attendre que la vérification se fasse.

Si l'on a effectué nos sauvegardes sur un support de stockage externe, il faut commencer par le brancher sur l'ordinateur.

La méthode évidente pour s'assurer que les sauvegardes sont toujours lisibles est sans doute de simuler une restauration. Procéder ainsi a un inconvénient… de taille : il faut avoir assez d'espace libre à notre disposition pour recopier l'ensemble des données sauvegardées vers un dossier temporaire que l'on supprime ensuite.

Voici une autre méthode, peut-être moins facile à mettre en œuvre, mais qui n'a pas cette contrainte. Elle nécessite d'utiliser un *Terminal*.

page 85

On va commencer la commande en tapant (**sans** faire *Entrée*) :

```
find
```

Ajouter ensuite un espace. Puis il faut indiquer le dossier contenant les sauvegardes, ce que l'on va faire avec la souris, en attrapant l'icône du dossier et en l'amenant sur le terminal. Après avoir relâché le bouton, ce qui est affiché doit ressembler à :

```
find '/media/externe/sauvegardes'
```

Il faut ensuite taper la fin de la commande pour que le tout ressemble à :

```
find '/media/externe/sauvegardes' -type f -print0 | xargs -0 cat > /dev/null
```

La lecture se lance dès qu'on a appuyé sur *Entrée*. La ligne suivante devrait rester vide jusqu'à la fin de l'opération.

Après de la patience et le retour du `$` de l'invite de commande, on peut fermer le terminal.

Si des messages d'erreurs sont apparus dans l'intervalle, tels que « *Erreur d'entrée/-sortie* » ou « *Input/output error* », cela indique que la sauvegarde est corrompue. En règle générale, il faut alors se débarrasser du support (CD ou DVD, clé USB ou disque dur), en prendre un autre et refaire une nouvelle sauvegarde.

Note : ces deux méthodes partagent le défaut de ne pas vérifier l'intégrité des données. page 41 Mettre en place un mécanisme pour le faire est difficile sans recourir à des logiciels de sauvegarde plus complexes.

19.2 En utilisant Déjà Dup

Durée : 5 minutes pour installer le logiciel.

On peut également préférer utiliser un logiciel spécialisé dans la réalisation des sauvegardes. L'un d'entre eux, baptisé « Déjà Dup », a comme avantages d'être facile à utiliser, et de réaliser des sauvegardes chiffrées. Ces sauvegardes sont également « incrémentales », c'est-à-dire que les fichiers inchangés depuis la sauvegarde précédente ne sont pas copiés une nouvelle fois, et qu'il est possible d'accéder aux fichiers tels qu'ils étaient à chacune des sauvegardes.

Ce qui le rend aussi simple peut être aussi une limite : il ne sait gérer qu'une seule configuration à la fois. On ne peut donc pas sauvegarder des dossiers différents sur des supports différents à des fréquences différentes. C'est surtout l'outil idéal pour sauvegarder l'essentiel du contenu de son dossier personnel, mais pas beaucoup plus.

Il n'est pas livré avec l'environnement par défaut, il est donc nécessaire d'installer le page 120 paquet Debian `deja-dup` pour pouvoir s'en servir.

19.2.1 Effectuer une sauvegarde

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 15 minutes environ pour la configuration, de quelques minutes à plusieurs heures pour la sauvegarde, selon la taille de ce qu'on veut copier.

On ouvre Déjà Dup via *Applications* → *Outils systèmes* → *Administration* → *Sauvegarder*.

L'interface est simplissime : deux énormes boutons nous accueillent, l'un pour *restaurer des fichiers d'une sauvegarde précédente*, l'autre pour *Afficher seulement les paramètres de sauvegarde*.

Avant que toute sauvegarde soit possible, il faut donc *Afficher* ces paramètres.

1. Dans le menu de gauche, cliquer sur *Stockage* pour choisir l'emplacement de la sauvegarde, ainsi que le nom du nouveau dossier où elle sera enregistrée.
2. Dans *Dossiers*, on trouve la liste des dossiers à inclure dans la sauvegarde (à gauche), et celle des dossiers à exclure (à droite). Laisser le *Dossier personnel* dans la fenêtre de gauche est suffisant dans la plupart des cas. Dans la liste de droite, on peut rajouter les dossiers contenant des fichiers souvent volumineux, comme *Vidéos* ou *Musique*.
3. Si l'on désire effectuer ces sauvegardes automatiquement, cliquer sur *Planification* pour choisir la fréquence de sauvegarde et la durée de conservation des données.

4. On peut maintenant retourner à la *Vue d'ensemble*. Dans le cas où l'on a choisi une sauvegarde automatique, déplacer le curseur sur la droite pour lancer cette dernière, sinon, cliquer sur *Sauvegarder maintenant*. Une nouvelle fenêtre s'ouvre, nous demandant une phrase de passe pour chiffrer [1] notre nouvelle sauvegarde. une fois la phrase de passe confirmée, cliquer sur *continuer* pour démarrer la sauvegarde.
5. On peut maintenant fermer Déjà Dup. Lors de la prochaine sauvegarde, si les paramètres ne sont pas changés, il suffira de donner la phrase de passe de la sauvegarde pour que cette dernière soit mise à jour.

page 91

On peut modifier plus tard tous ces paramètres en redémarrant Déjà Dup.

Lorsque la planification des sauvegardes est activée et que le temps indiqué depuis la précédente sauvegarde est écoulé, Déjà Dup affiche un message de notification pour nous signifier qu'il effectuera la prochaine sauvegarde dès que le support externe sera de nouveau branché sur l'ordinateur. Et dès que ce sera le cas, une fenêtre s'ouvrira automatiquement pour demander de saisir la *phrase de passe* nécessaire pour mettre à jour la sauvegarde.

19.2.2 Restaurer une sauvegarde

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 5 minutes pour configurer, de quelques minutes à plusieurs heures pour la restauration, selon la taille de nos sauvegardes.

On ouvre Déjà Dup via *Applications* → *Outils systèmes* → *Administration* → *Sauvegarde*. L'opération de restauration démarre simplement en cliquant sur le bouton *Restaurer*.

Si c'est la première fois qu'on utilise Déjà Dup (par exemple pour restaurer son dossier personnel après la perte d'un disque dur), il nous demande d'indiquer le dossier où ont été effectuées les sauvegardes. Sinon, il utilise le dossier déjà configuré.

Après un court délai, Déjà Dup nous demande de choisir, avec sa date, la sauvegarde à restaurer.

Il faut ensuite indiquer le dossier où seront écrits les fichiers issus de la sauvegarde. On peut soit restaurer vers l'emplacement original (ce qui remplace éventuellement des fichiers par la version qui se trouvait dans la sauvegarde), soit indiquer un autre dossier.

Pour finir, un dernier écran résume les paramètres de cette restauration. Après avoir cliqué sur *Restaurer*, une fenêtre s'ouvre, si nécessaire, pour demander le mot de passe du superutilisateur (par exemple, pour restaurer les permissions de certains fichiers). Si la sauvegarde a été chiffrée, Déjà Dup nous demande ensuite la phrase de passe. Une fois que l'on aura cliqué sur *Continuer*, l'écriture des fichiers en provenance de la sauvegarde commencera pour de bon.

19.2.3 S'assurer que les sauvegardes sont toujours lisibles

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

1. Si le support externe est chiffré, on peut éventuellement décider de ne pas chiffrer les fichiers sauvegardés. Cela fait une phrase de passe de moins à inventer et à retenir. On perd néanmoins la possibilité de compartimenter les accès, au cas où le support externe servirait à d'autres choses que les sauvegardes.

Durée : de quelques minutes à plusieurs heures, selon la taille de nos sauvegardes.

Le fonctionnement incrémental de Déjà Dup assure déjà superficiellement que les sauvegardes précédentes soient lisibles. Néanmoins cela ne constitue pas une garantie…

Malheureusement, la meilleure méthode actuellement disponible avec Déjà Dup pour s'assurer que l'on peut restaurer ses sauvegardes est… de faire une restauration vers un dossier temporaire que l'on effacera après. C'est loin d'être pratique, vu qu'il faut avoir accès à un disque dur chiffré suffisamment grand.

On peut toutefois s'assurer que les fichiers contenant les sauvegardes restent lisibles en utilisant les mêmes méthodes que celles décrites précédemment. page 144

Chapitre 20

Partager un secret

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : une heure environ.

Parfois, on souhaite être plusieurs à partager un secret, sans pour autant que chaque personne ne dispose de la totalité du secret.

Cela tombe bien, plusieurs techniques cryptographiques ont été inventées pour cela. Elles permettent toutes, mais avec des calculs mathématiques un peu différents, de découper un secret en plusieurs morceaux, que l'on pourra reconstituer en en réunissant quelques-uns [1].

20.1 Partager une phrase de passe

L'usage le plus pratique est de partager comme secret la phrase de passe d'un support chiffré. page 137

Cette étape doit idéalement être faite à partir d'un système *live* afin de ne pas laisser des traces du secret que l'on va partager. page 99

20.1.1 Installer le paquet nécessaire

Pour réaliser le partage du secret, on utilisera le programme `ssss-split`. On trouve ce programme parmi ceux fournis par le système live *Tails*, cependant, pour en disposer sur une Debian chiffrée, il est nécessaire d'installer le paquet Debian *ssss*. page 120

Les outils contenus dans le paquet *ssss* sont à utiliser en ligne de commande. Toutes les opérations devront donc être effectuées dans un *Terminal*, sans les pouvoirs d'administration. page 85

20.1.2 Générer une phrase de passe aléatoire

Dans notre cas, personne ne doit pouvoir ni se souvenir ni deviner la phrase de passe qui sera utilisée pour le chiffrement. On va donc générer une phrase de passe complètement aléatoire en tapant la commande :

```
head -c 32 /dev/random | base64
```

1. Pour plus de détails, voir l'article de Wikipédia sur les secrets répartis [https://fr.wikipedia.org/wiki/Secret_réparti].

L'ordinateur va répondre quelque chose comme :

```
7rZwOOu+8v1stea98OuyU1efwNzHaKX9CuZ/TK0bRWY=
```

Si l'on désire une phrase de passe de plus ou moins de 32 caractères, il suffira de remplacer 32 par le nombre de caractères désirés. Sélectionner cette ligne à l'aide de la souris et la copier dans le presse-papiers (via le menu *Édition* → *Copier*).

20.1.3 Découper le secret

Avant de découper le secret, il faut décider en combien de morceaux il sera découpé, et combien de morceaux seront nécessaires pour le reconstituer.

Ensuite, toujours à l'aide de notre terminal, il faut utiliser `ssss-split` de la façon suivante :

```
ssss-split -t NOMBRE_DE_MORCEAUX_NECESSAIRES -n NOMBRE_DE_MORCEAUX_TOTAL
```

Le `NOMBRE_DE_MORCEAUX_NECESSAIRES` est le nombre de morceaux qu'il sera nécessaire de réunir pour retrouver la phrase de passe de départ. Le `NOMBRE_DE_MORCEAUX_TOTAL` correspond au nombres de morceaux en lesquels la phrase de passe sera découpée. Le message `WARNING: couldn't get memory lock` peut être ignoré sans problème si on utilise bien un système *live*.

Lorsqu'il demande le secret, on peut coller le contenu du presse-papier, à l'aide du menu *Édition* → *Coller*. Appuyer ensuite sur la touche *Entrée* pour valider la commande.

Chaque personne partageant le secret devra conserver l'une des lignes affichées ensuite. Cela dans leur **intégralité**, en prenant également bien en note le premier chiffre suivi du tiret.

Voici un exemple avec la clé aléatoire générée précédemment, partagée entre 6 personnes et qui nécessitera que 3 d'entre elles se réunissent pour la retrouver :

```
$ ssss-split -t 3 -n 6
Generating shares using a (3,6) scheme with dynamic security level.
Enter the secret, at most 128 ASCII characters :Using a 352 bit security level.
1-b8d576a1a8091760b18f125e12bb6f2b1f2dd9d93f7072ec69b129b27bb8e97536ea85c7f6dcee7b
↪    4399ea49
2-af83f0af05fc207e3b466caef30ec4d39c060800371feab93594350b7699a8db9594bfc71ed9cd2b
↪    f314b738
3-4718cb58873dab22d24e526931b061a6ac331613d8fe79b2172213fa767caa57d29a6243ec0e6cf7
↪    7b6cbb64
4-143a1efcde7f4f5658415a150fcac6da04f697ebfeb9427b59dca57b50ec755510b0e57ccc594e6b
↪    1a1eeb04
5-fca1250b5cbec40ab14964d2cd7463af34c389f81158d1707b6a838a500977d957be38f83e8eefb7
↪    9266e74a
6-ebf7a305f14bf3143b801a222cc1c857b7e8582119374925274f9f335d283677f4c002f8d68bcce7
↪    22ebba1f
```

20.1.4 Créer le support chiffré

page 137

On pourra ensuite créer le support chiffré. Au moment d'indiquer la phrase de passe, on pourra copier le contenu du presse-papier, comme précédemment, ou alors la retranscrire en l'ayant sous les yeux.

20.2 Reconstituer la phrase de passe

Afin de reconstituer la phrase de passe, il est nécessaire de disposer d'au moins autant de morceaux que le nombre minimal décidé lors du découpage.

Cette étape doit idéalement être faite à partir d'un système *live* afin de ne pas laisser de traces du secret partagé. page 99

20.2.1 Installer les paquets nécessaires

Comme précédemment, si le programme n'est pas disponible sur le système on aura besoin d'avoir installé le paquet *ssss* et d'avoir ouvert un terminal. page 120

20.2.2 Recombiner le secret

Afin de recombiner le secret, on utilisera le programme `ssss-combine`. Il est nécessaire de lui indiquer le nombre de morceaux qu'on a à notre disposition :

```
ssss-combine -t NOMBRE_DE_MORCEAUX_A_DISPOSITION
```

Le programme demande ensuite de saisir les morceaux à notre disposition. Il faut taper *Entrée* après chacun d'entre eux. Si tout se passe bien, le programme affichera ensuite la phrase de passe complète.

Pour reprendre l'exemple précédent, cela donne :

```
$ ssss-combine -t 3
Enter 3 shares separated by newlines :
Share [1/3] :
↪    4-143a1efcde7f4f5658415a150fcac6da04f697ebfeb9427b59dca57b50ec755510b0e5
↪    7ccc594e6b1a1eeb04
Share [2/3] :
↪    2-af83f0af05fc207e3b466caef30ec4d39c060800371feab93594350b7699a8db9594bf
↪    c71ed9cd2bf314b738
Share [3/3] :
↪    6-ebf7a305f14bf3143b801a222cc1c857b7e8582119374925274f9f335d283677f4c002
↪    f8d68bcce722ebba1f
Resulting secret :7rZwOOu+8v1stea98OuyU1efwNzHaKX9CuZ/TK0bRWY=
```

Attention, si un des morceaux a mal été tapé, l'erreur qui s'affiche n'est pas forcément très explicite :

```
$ ssss-combine -t 3
Enter 3 shares separated by newlines :
Share [1/3] :
↪    4-143a1efcde7f4f5658415a150fcac6da04f697ebfeb9427b59dca57b50ec755510b0e5
↪    7ccc594e6b1a1eeb04
Share [2/3] :
↪    2-af83f0af05fc207e3b466caef30ec4d39c060800371feab93594350b7699a8db9594bf
↪    c71ed9cd2bf31ab738
Share [3/3] :
↪    6-ebf7a305f14bf3143b801a222cc1c857b7e8582119374925274f9f335d283677f4c002
↪    f8d68bcce722ebba1f
Resulting secret :......L.fm.....6 _....v..w.a....[....zS.....
WARNING :binary data detected, use -x mode instead.
```

20.2.3 Ouvrir le support chiffré

Une fois la phrase de passe obtenue, on peut utiliser un copier/coller afin de déverrouiller le support chiffré, ou alors la retranscrire en l'ayant sous les yeux.

Chapitre 21

Utiliser les sommes de contrôle

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 5 à 10 minutes.

Dans la première partie, on a évoqué les *sommes de contrôle*, des « nombres » qui permettent de vérifier l'intégrité d'un fichier (ou de toutes autres données). Le principe est qu'il est quasiment impossible d'avoir une somme de contrôle identique pour deux fichiers différents. Si Alice dit à Betty dans une lettre que le programme que cette dernière peut télécharger sur son site a pour somme de contrôle SHA256 `171a0233a4112858db23621dd5ffa31d269cbdb4e75bc206ada58ddab444651f` et que le fichier qu'elle a reçu a la même somme de contrôle, il est quasiment sûr que personne n'a falsifié le programme en chemin, et elle peut exécuter le programme sans trop de craintes.

page 41

Il existe plusieurs algorithmes pour faire des sommes de contrôles. Parmi eux :

- MD5 n'est plus sûr de nos jours et est à proscrire ;
- SHA1 est très utilisé, mais est en voie d'être cassé. Il faut l'abandonner ;
- SHA224, SHA256, SHA384 et SHA512 sont pour l'instant toujours sûrs. Nous allons utiliser SHA256, mais les mêmes méthodes fonctionnent avec les autres algorithmes.

21.1 Obtenir la somme de contrôle d'un fichier

Que l'on souhaite vérifier l'intégrité d'un fichier, ou permettre à son correspondant de le faire, il faut calculer la somme de contrôle de ce fichier.

page 41

Il est possible d'utiliser un outil graphique tout aussi bien qu'un terminal pour effectuer de tels calculs, nous n'allons cependant pas détailler l'utilisation d'un terminal.

21.1.1 Installer les logiciels nécessaires

Si le paquet `nautilus-gtkhash` n'est pas encore installé, l'installer. Ce paquet est installé par défaut dans *Tails*.

page 120

21.1.2 Utiliser GtkHash

Pour lancer *GtkHash*, aller dans *Applications* → *Accessoires* → *GtkHash*. Une fenêtre s'ouvre proposant de sélectionner un fichier pour lequel des sommes de contrôle seront calculées, avec trois sélections par défaut, MD5, SHA1, SHA256. Si une autre somme de contrôle que celles-ci est nécessaire, aller dans *Édition* → *Préférences* et cocher les

sommes de contrôles désirées pour les voir s'afficher. Une fois cela terminé, cliquer sur le champ contenant *(Aucun)* situé après l'entrée *Fichier* et sélectionner le fichier dont on veut calculer une somme de contrôle. Cliquer enfin sur *Hash*, les sommes de contrôles apparaissent alors.

21.2 Vérifier l'intégrité d'un fichier

Il faut obtenir la somme de contrôle du fichier original par un moyen sûr, autre que celui par lequel on a reçu le fichier. Par exemple, si l'on a téléchargé le fichier, on peut avoir reçu sa somme de contrôle dans une lettre, ou par téléphone — le mieux étant bien sûr de vive voix.

De la même manière, pour permettre à d'autres personnes de vérifier l'intégrité d'un fichier, on leur fera parvenir la somme de contrôle selon les mêmes méthodes.

Grâce à l'une des méthodes ci-dessus, obtenir la somme de contrôle de sa copie du fichier. Prendre garde à utiliser le même algorithme que celui qui a été utilisé par son correspondant. Si l'on utilise SHA1 et qu'il utilise SHA256, on n'aura bien sûr pas la même somme de contrôle. Si notre correspondant nous propose plusieurs sommes de contrôle, préférer l'algorithme le plus dur à casser.

page préc.

Vérifier que les deux sommes de contrôle sont les mêmes — c'est un peu long et fastidieux, mais c'est souvent plus simple à deux, ou en les collant l'une en-dessous de l'autre dans un fichier texte.

Chapitre 22

Installer et utiliser un système virtualisé

Cet ensemble de recettes traite de l'utilisation d'un système d'exploitation virtuel à l'intérieur d'un système GNU/Linux. Elles sont utilisées par le cas d'usage qui parle de travailler sur un document sensible sous Windows.

page 68

22.1 Installer VirtualBox

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : un quart d'heure environ.

22.1.1 Principe

L'objectif de cette recette est d'installer VirtualBox, un logiciel qui permet de faire tourner un système d'exploitation (appelé invité) à l'intérieur d'un autre (appelé hôte) : on appelle cela de la *virtualisation*. Cette technologie, ainsi qu'une politique de sécurité l'utilisant, est décrite plus avant dans le cas d'usage expliquant comment travailler sur un document sensible sous Windows.

page 68

22.1.2 Installer VirtualBox

page 120

L'étape suivante est donc d'installer les paquets `virtualbox-qt` et `virtualbox-dkms`.

22.1.3 Vérifier l'installation

Lancer VirtualBox à partir du menu *Applications* → *Accessoires* → *VirtualBox*. Une fenêtre s'ouvre et nous souhaite la bienvenue dans VirtualBox. Refermons-la, car nous avons encore quelques préparatifs à faire avant de nous servir de ce logiciel.

22.1.4 Créer un dossier pour sauvegarder les images propres

page 68

Comme expliqué dans le cas d'usage, on aura plus tard envie de sauvegarder des images de systèmes propres. Créons dès maintenant un dossier pour cela, par exemple en ajoutant dans le *Dossier personnel* (accessible depuis le menu *Raccourcis*) un dossier *Disques virtuels propres*.

22.2 Installer un Windows virtualisé

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 20 minutes environ, plus le temps d'installer Windows (de 30 minutes à plus d'une heure).

Avant tout chose, se munir d'un CD d'installation de la version de Windows appropriée, et l'insérer dans le lecteur CD/DVD. Si une fenêtre affichant le contenu du CD s'ouvre automatiquement, la refermer ou l'ignorer.

22.2.1 Préparer l'installation sur VirtualBox

Depuis le bureau, aller dans *Applications* → *Accessoires* → *VirtualBox*.

Le programme démarre. Cliquer sur *New* et suivre l'assistant :

- Choisir un *Nom* pour la machine virtuelle.
- Choisir le *Type de l'OS* correspondant parmi les versions de Windows proposées.
- Indiquer la taille de *Mémoire* vive dédiée à la machine virtuelle. Pour Windows, 512 Mio est le minimum recommandé.
- Créer un *Disque dur virtuel* pour accueillir le système d'exploitation virtuel :
  - cocher *Disque d'amorçage* et *Créer un nouveau disque dur* ;
  - une fenêtre s'ouvre, cliquer sur *Suivant* ;
  - choisir *Dynamiquement alloué* (l'image disque s'agrandira au besoin, jusqu'à atteindre au maximum la taille indiquée) ;
  - dans *Emplacement*, donner un nom au fichier image disque (on peut aussi choisir son emplacement en cliquant sur le petit dossier à droite de cette ligne, mais c'est bien de le laisser à l'endroit suggéré) ;
  - choisir la *Taille* de l'image virtuelle : sachant qu'on veut accueillir tout un Windows, elle doit être conséquente ! 20 Go, c'est bien si on a assez de place ; en cas de petit disque dur, essayer moins…
  - cliquer sur *Suivant* puis *Créer* : le logiciel a créé et sélectionné un disque dur virtuel.
- Cliquer sur *Créer*.

Sur la fenêtre principale de VirtualBox, la liste contient maintenant notre nouvelle machine virtuelle. Il reste à y installer le système d'exploitation…

Mais avant tout, on va cliquer sur le bouton *Configuration* : ce menu permettra par la suite de la configurer. Pour l'instant on va juste lui dire deux choses :

1. Il faut couper l'accès au réseau (pour des raisons de sécurité déjà mentionnées) :
   - aller dans la catégorie *Réseau* ;
   - décocher la case *Activer la carte réseau* dans tous les onglets où elle est déjà cochée par défaut (généralement, dans un seul : le premier).
2. Il faut lancer le système sur le CD d'installation de Windows qu'on a mis dans le lecteur CD/DVD :
   - aller dans la catégorie *Stockage* ;
   - dans *Arborescence Stockage*, sélectionner la ligne contenant une icône de CD et intitulée *Vide* ;
   - ensuite, dans *Attributs*, à la ligne *Lecteur CD/DVD*, sélectionner *Lecteur de l'hôte […]*, ce qui correspond à votre lecteur CD/DVD habituel.

Cliquer ensuite sur *OK* pour enregistrer les paramètres.

22.2.2 Lancer la machine virtuelle

Si ce n'est pas déjà fait, c'est le moment de glisser le CD d'installation de Windows dans le lecteur de l'ordinateur.

En ayant toujours notre nouvelle machine virtuelle sélectionnée, on peut maintenant cliquer sur l'icône *Démarrer*. Le système virtuel démarre… c'est le moment de découvrir l'utilisation de la machine virtuelle.

Lorsqu'elle est lancée, la machine virtuelle fonctionne dans une fenêtre qui permet de gérer son utilisation :

- en haut à gauche : une barre de menus contenant *Machine*, *Écran*, *Périphériques*, *Aide* ;
- en bas à droite : des icônes indiquant comment la machine virtuelle utilise le matériel. On peut par exemple vérifier, en passant la souris dessus, le comportement du curseur de souris entre système hôte et invité.

Au premier clic dans la fenêtre, le logiciel explique qu'il va capturer la souris ; à la première touche tapée, il explique qu'il capture le clavier. Il faut bien prendre en compte ce qu'il indique, c'est ce qui permet de savoir comment par la suite sortir de la machine virtuelle !

Enfin, tout ça est expliqué par le logiciel. Il nous reste donc à installer le Windows virtuel.

22.2.3 Installer Windows

Le système virtuel démarre sur le lecteur CD/DVD qu'on lui a indiqué et commence l'installation.

On ne rentrera pas dans les détails du processus. On peut toutefois préciser :

- Au moment de formater la partition, mieux vaut choisir *Formater avec NTFS (rapide)*.
- Ne pas mettre d'informations personnelles lorsque le *Nom* et l'*Organisation* sont demandés. Mettre par exemple un simple point (« . ») dans les cases permet, la plupart du temps, de continuer l'installation.
- Lors de la configuration du réseau, un message d'erreur peut être affiché. C'est bon signe : nous avons désactivé le réseau de la machine virtuelle.

22.2.4 Démarrer sur le système invité

Une fois l'installation terminée, éteindre la machine virtuelle. Retourner dans la fenêtre principale de VirtualBox. Sélectionner la machine virtuelle qu'on vient d'installer et cliquer sur l'icône *Configuration*. Dans la liste de gauche, choisir *Stockage*. Dans *Arborescence Stockage*, sélectionner la ligne contenant une icône de CD et dans la liste déroulante *Lecteur CD/DVD* choisir *Ejecter le disque du lecteur virtuel*. Fermer alors la fenêtre de configuration.

Démarrer alors la machine virtuelle en cliquant sur *Démarrer*.

22.2.5 Installer les logiciels bonus pour système invité

Dans la fenêtre qui accueille Windows, ouvrir le menu *Périphériques* qui propose d'*Installer les Additions invité…*. Si ça n'a pas été fait auparavant, VirtualBox proposera de télécharger l'image ISO qui les contient. Une barre de progression en bas à gauche de la fenêtre indique que le téléchargement est en cours, puis VirtualBox demande *Voulez-vous monter ce CD dans le lecteur virtuel*. Cliquer sur *Insérer*. Un

nouveau CD-ROM est maintenant ajouté à l'environnement de Windows. Si le programme d'installation ne se lance pas automatiquement, il faut lancer *VBoxWindowsAdditions* qui se trouve sur ce nouveau CD-ROM. Reste ensuite à accepter les choix par défaut pour installer les « Additions invité ».

Une nouvelle icône à l'aspect de cube transparent apparaît alors après redémarrage de la machine virtuelle en bas à droite du bureau Windows. Elle signifie que les « Additions » on été installées.

Éteindre le Windows virtuel. L'installation du Windows virtuel est maintenant terminée.

22.3 Sauvegarder une image de disque virtuel propre

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 10 minutes.

page 68 Comme indiqué dans la méthode permettant de travailler sur un document sensible sous Windows, on peut avoir besoin de sauvegarder l'image disque d'une machine virtuelle.

22.3.1 Éteindre la machine virtuelle

Si la machine virtuelle propre, qui doit être sauvegardée, est en cours d'utilisation, il faut commencer par l'éteindre (par exemple via le menu *Machine* → *Fermer…* → *Envoyer le signal d'extinction* de VirtualBox, ou, tout bonnement, depuis le menu *Démarrer* de windows).

22.3.2 Ouvrir le dossier des disques virtuels de VirtualBox

Dans le *Dossier personnel*, ouvrir le dossier *VirtualBox VMs*.

22.3.3 Effectuer la sauvegarde

- Sélectionner le disque virtuel dont le nom correspond à celui de la machine virtuelle, par exemple `Windows 2000.vdi`.
- Dans le menu *Édition* choisir *Copier*.
- page 156 Aller dans le dossier de sauvegarde des images propres. Si l'on a suivi les conseils donnés précédemment, il s'agit du dossier *Disques virtuels propres* dans *Dossier personnel*.
- Dans le menu *Édition* choisir *Coller* pour obtenir une copie du fichier.
- Sélectionner la copie, et la renommer à partir du menu *Édition* → *Renommer…*. Entrer un nouveau nom, par exemple `Sauvegarde propre de Windows 2000.vdi`.

22.3.4 Effacer la machine virtuelle

page ci-contre On ne va plus se servir de cette machine propre. C'est donc le moment de suivre la recette sur l'effacement d'une machine virtuelle.

22.4 Effacer « pour de vrai » une machine virtuelle

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 10 minutes pour supprimer + quelques minutes à quelques heures pour écraser les données.

Cette recette vise à effacer proprement une machine virtuelle.

22.4.1 Supprimer la machine virtuelle de VirtualBox

Ouvrir la fenêtre principale de VirtualBox, accessible depuis le menu *Applications* → *Accessoires*.

Sélectionner la machine virtuelle à effacer.

Dans le menu *Machine* choisir *Supprimer*, puis choisir *Supprimer de la liste*.

22.4.2 Effacer le disque dur virtuel et son contenu

Ouvrir le dossier des disques virtuels de VirtualBox, *VirtualBox VMs* depuis *Raccourcis* → *Dossier personnel*. Sélectionner le dossier contenant la machine virtuelle à effacer puis utiliser l'outil effacer des fichiers avec leur contenu pour effacer le disque virtuel de la machine virtuelle en question.

page 129

22.4.3 Prévenir VirtualBox que le disque virtuel n'existe plus

Dans le menu *Fichier* de VirtualBox, ouvrir le *Gestionnaire de médias…* puis dans l'onglet *Disques durs*, sélectionner la ligne correspondant au fichier effacé (précédée d'un panneau jaune quelque peu alarmiste), et cliquer sur le bouton *Supprimer*.

Le gestionnaire de médias virtuels, ainsi que la fenêtre de VirtualBox, peuvent maintenant être refermés.

22.5 Créer une nouvelle machine virtuelle à partir d'une image propre

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : selon la taille du disque.

L'objectif de cette recette est de « décongeler » une image de disque virtuel propre préalablement sauvegardée, afin de l'utiliser pour un nouveau projet, comme le recommande la méthode préconisée pour travailler sur un document sensible sous Windows.

page 160 page 68

22.5.1 Choix du nom

Il faudra choisir un nom pour cette nouvelle machine virtuelle et les fichiers qui lui correspondent. Ces fichiers étant situés sur le système hôte, ce nom laissera quasi inévitablement des traces dessus, même une fois la machine virtuelle supprimée. Il s'agit donc de choisir ce nom en connaissance de cause.

22.5.2 Copier l'image de disque virtuel

On ne peut pas simplement copier le fichier congelé, car VirtualBox se plaindrait qu'on a deux disques virtuels identiques. Il existe cependant une commande pour recopier un disque virtuel, mais elle n'est accessible que depuis la ligne de commande.

page 85

Commençons donc par ouvrir un terminal (*Applications* → *Accessoires* → *Terminal*).

Ensuite, recopions l'image précédemment décongelée avec la commande :

```
VBoxManage clonehd SAUVEGARDE NOUVEAU_DISQUE
```

Nous allons à présent voir comment construire cette ligne de commande, car il faut remplacer `SAUVEGARDE` par le chemin d'accès à la sauvegarde du disque virtuel, et `NOUVEAU_DISQUE` par le chemin du nouveau disque.

Attention ! Si l'on souhaite taper les noms de fichiers à la main, il faut savoir que les chemins sont relatifs au dossier de VirtualBox — si l'on ne change pas les options, il s'agit de `.VirtualBox`. Pour corriger cela, on pourra par exemple mettre des chemins absolus.

La façon la plus simple de faire est de commencer par taper :

```
VBoxManage clonehd
```

Ensuite, il faut ajouter un espace. Puis avec la souris, attraper l'icône du disque virtuel à décongeler (dans le dossier *Disques virtuels propres*) et la lâcher au-dessus du terminal.

Pour ajouter le nouveau disque, on recommence l'opération avec l'icône du dossier *VirtualBox VMs*.

L'affichage devrait à présent ressembler à :

```
VBoxManage clonehd '/home/LOGIN/Disques virtuels propres/Sauvegarde propre de
↪    Windows XP.vdi' '/home/LOGIN/VirtualBox VMs'
```

Un espace a été ajouté automatiquement avec l'insertion du chemin. On va le supprimer, pour ajouter ensuite le nom du nouveau disque, en écrivant par exemple `/Projet1/Projet1.vdi`.

Au final, cela doit ressembler d'assez près à :

```
VBoxManage clonehd '/home/LOGIN/Disques virtuels propres/Sauvegarde propre de
↪    Windows 2000.vdi' '/home/LOGIN/VirtualBox VMs/Projet1/Projet1.vdi'
```

Après toutes ces étapes, la ligne de commande est complète, et on peut lancer son exécution en tapant sur la touche *Entrée*.

22.5.3 Créer une nouvelle machine virtuelle

Dans le bureau Debian, aller dans *Applications* → *Accessoires* → *VirtualBox*.

Le programme démarre. Cliquer sur *New* et suivre l'assistant :

- choisir un *Nom* pour la machine virtuelle ;
- choisir le *Type de l'OS* correspondant parmi les Windows proposés ;
- choisir la taille de *Mémoire* vive dédiée à la machine virtuelle, en fonction de la quantité dont on a besoin pour le projet prémédité : si on veut utiliser un gros logiciel comme Photoshop, il faut en prévoir le plus possible (au moins 512 Mo), en sachant que VirtualBox râlera si l'on attribue plus de la moitié de la mémoire totale à la machine virtuelle ;
- choisir *Disque d'amorçage* et *Utiliser un disque dur existant*. Cliquer sur l'icône à droite du menu déroulant pour ouvrir une fenêtre qui nous permettra de sélectionner notre image *Projet1.vdi*. Cliquer sur *Ouvrir* ;
- cliquer sur *Suivant* et *Créer*.

Il faut maintenant **avant tout** configurer la machine virtuelle. Cliquer sur le bouton *Configuration* en prenant soin de la sélectionner dans la liste auparavant.

Il faut couper l'accès au réseau (pour des raisons de sécurité déjà mentionnées) :

- Aller dans la section *Réseau* ;
- Décocher la case *Activer la carte réseau* dans tous les onglets où elle est déjà cochée par défaut (généralement, dans un seul : le premier).

Cliquer ensuite sur *OK* pour enregistrer les paramètres.

Créer un compte « utilisateur » pour le nouveau projet

Comme expliqué dans le cas d'usage, on souhaite travailler sur un compte « utilisateur » différent pour chaque projet. Voici comment le faire avec Windows XP — ça ne doit pas être trop différent avec d'autres versions.

page 68

Démarrer la nouvelle machine virtuelle en cliquant sur *Démarrer*.

Une fois dans le Windows virtualisé, ouvrir *Démarrer* → *Panneau de configuration* puis choisir *Comptes d'utilisateurs* et *Créer un nouveau compte*.

Choisir alors un nom pour le nouveau compte, tout en gardant à l'esprit que ce nom sera probablement enregistré dans les documents créés. Choisir ensuite de créer un compte *Administrateur de l'ordinateur*[1] et cliquer sur *Créer un compte*.

Fermer alors la session à partir du menu *Démarrer*. On veillera à ne plus utiliser pour ce projet que le compte nouvellement créé.

1. Étant donné que l'on utilise un disque virtuel propre pour chaque projet et que l'on a pas accès au réseau, cela ne constitue pas un grand risque et nous simplifiera la vie.

22.6 Envoyer des fichiers à un système virtualisé

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 10 minutes environ.

Vu que le Windows *invité* n'a pas le droit de sortir de sa boîte pour aller chercher lui-même des fichiers, il peut être nécessaire de lui en faire parvenir depuis « l'extérieur ». Voyons donc comment procéder.

22.6.1 Depuis un CD ou DVD

C'est nécessaire si on veut installer d'autres logiciels sous Windows virtuel :

- Insérer le CD à lire dans le lecteur, attendre quelques secondes, puis reprendre le contrôle avec le système hôte (Ctrl + ou *Home*), et dans la fenêtre accueillant Windows, cliquer sur *Périphérique* puis *Lecteurs CD/DVD* et sélectionner *Lecteur de l'hôte […]*.
- Windows devrait alors détecter le CD inséré. Si ce n'est pas le cas, on peut aller le chercher dans *Menu Démarrer → Poste de travail*. Si ça ne marche pas du premier coup, recommencer l'opération.

On peut ainsi charger les logiciels depuis le lecteur CD de l'ordinateur : il seront installés durablement sur le disque dur virtuel.

22.6.2 Depuis un dossier

Il est possible de rendre un dossier du système hôte lisible par Windows. Mais veillons à ce que ce ne soit pas n'importe quel dossier…

Créer un dossier réservé à cet effet dans le système hôte

Réduire la fenêtre accueillant le système invité. Ensuite, choisir l'emplacement où on veut mettre ce dossier d'échange. Par exemple : dans le *Dossier personnel* faire un clic-droit puis *Créer un dossier* et lui donner un nom évocateur (« *Dossier lisible par Windows* », par exemple).

Indiquer au gestionnaire de la machine virtuelle où se trouve ce dossier

Aller dans la fenêtre de VirtualBox dans laquelle est lancée la machine virtuelle Windows et ouvrir le menu *Périphériques → Dossiers partagés…* Ajouter un dossier en cliquant sur l'icône avec un « + » en haut à droite. Une boîte de dialogue s'ouvre :

- dans *Chemin du dossier* cliquer sur *Autre…* et indiquer l'emplacement du dossier à partager ;
- dans *Nom du dossier*, le nom que le dossier aura à l'intérieur du système virtuel s'affiche, par exemple « *partage* ». Il est possible de le modifier, mais ce nom doit être court, et ne doit pas contenir d'espace ;
- cocher la case *Lecture seule*. Ainsi, le système virtuel ne pourra que lire le contenu du dossier, mais rien y écrire ;
- si, et seulement si, le partage de ce dossier doit être permanent, sélectionner *Configuration permanente* ; sinon, le partage ne sera activé que pour cette session.

Attention : avant de valider, il faut être bien sûr que l'on veut laisser le système Windows lire tout le contenu du dossier qu'on a demandé de partager. Si c'est bon, cliquer sur *OK*, et refermer la fenêtre avec *OK*.

Indiquer à Windows XP où se connecter pour trouver ce dossier partagé

- Dans le menu *Démarrer*, ouvrir le *Poste de travail*.
- Dans le menu *Outil*, cliquer sur *Connecter un lecteur réseau*.
- Windows propose un nom de lecteur (par exemple `Z:`) et demande d'indiquer le dossier : cliquer sur parcourir (à droite) → *VirtualBox Shared Folders* → *\\Vboxsvr* → *Nom_du_repertoire*, puis *OK*. On peut choisir au passage si on veut que ce « lecteur » ne soit accessible que pour la durée de la session en cours, ou à chaque nouvelle session.

Attention : en apprenant à utiliser ce système de partage, on pourrait être tenté de le configurer pour donner accès directement aux périphériques branchés sur le système hôte : c'est bien **la pire idée qu'on puisse avoir**, qui anéantirait à elle seule toute la politique de sécurité.

22.7 Faire sortir des fichiers d'un système virtualisé

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 10 minutes environ.

Le Windows *invité* n'a pas le droit, par défaut, de laisser des traces en dehors de son compartiment étanche. Mais presque inévitablement vient le temps où il est nécessaire d'en faire sortir des fichiers. Voyons donc comment procéder.

22.7.1 En gravant un CD ou DVD

Avant tout, sortir les CD ou les DVD qui pourraient être dans les lecteurs et auxquels on ne veut pas donner accès à la machine virtuelle.

Si la machine virtuelle est en fonction, l'éteindre.

Aller alors dans la fenêtre principale de VirtualBox et sélectionner dans la liste de gauche la machine virtuelle sur laquelle se trouvent les données à graver. Cliquer alors sur l'icône *Configuration*.

Dans la boîte de dialogue *Configuration*, sélectionner *Stockage* dans la liste de gauche, et, dans la section *Arborescence Stockage*, cliquer sur la ligne qui commence par une icône de CD. Dans *Lecteur CD/DVD* choisir *Lecteur de l'hôte* et cocher *Mode direct*. Valider en cliquant sur *Ok*.

Il est alors possible de relancer la machine virtuelle, et de graver les données depuis l'intérieur.

22.7.2 Dans un dossier vide

Il est possible de permettre à Windows d'écrire dans un dossier du système hôte. Mais veillons à ce que ce ne soit pas n'importe quel dossier…

Attention : en apprenant à utiliser ce système de partage, on pourrait être tenté de le configurer pour donner accès directement aux périphériques branchés sur le système hôte : c'est bien **la pire idée qu'on puisse avoir**, qui anéantirait à elle seule toute la politique de sécurité.

Créer un dossier réservé à cet effet dans le système hôte

- Réduire la fenêtre accueillant le système invité.
- Choisir l'emplacement où on veut mettre ce dossier d'échange. Par exemple, dans le *Dossier personnel*, faire un clic-droit puis *Créer un dossier* et lui donner un nom évocateur, comme « *Dossier où Windows peut écrire* ».

Indiquer au gestionnaire de la machine virtuelle où se trouve ce dossier

- Si la machine virtuelle est éteinte, la *démarrer*.
- Aller dans la fenêtre de VirtualBox dans laquelle est lancée la machine virtuelle Windows et ouvrir le menu *Périphériques* → *Dossiers partagés…*
- Ajouter un dossier en cliquant sur l'icône avec un « + » en haut à gauche. Une boîte de dialogue s'ouvre :
  - dans *Chemin du dossier* cliquer sur *Autre…* et indiquer l'emplacement du dossier à partager ;
  - dans *Nom du dossier*, le nom que le dossier aura à l'intérieur du système virtuel s'affiche. Choisir un nom court sans espace, par exemple « *sortie* » ;

– Si on veut exporter un dossier de façon permanente (et non pas pour cette session uniquement) cocher la case *Configuration permanente* ;
– ne **pas** cocher la case *Lecture seule*.

Attention : avant de valider, il faut être bien sûr que le dossier en question est vide. Windows pourra en effet non seulement y écrire, mais aussi y lire. Si c'est bon, cliquer sur *OK*, et refermer la fenêtre avec *OK*.

Indiquer à Windows XP où se connecter pour trouver ce dossier partagé

- Dans le menu *Démarrer*, ouvrir le *Poste de travail*.
- Dans le menu *Outils*, cliquer sur *Connecter un lecteur réseau*.
- Windows propose un nom de lecteur (par exemple `Z:`) et demande d'indiquer le dossier : cliquer sur parcourir (à droite) → *VirtualBox Shared Folders* → *\\Vboxsvr* → *Nom_du_dossier*, puis *OK*. On peut choisir au passage si on veut que ce « lecteur » ne soit accessible que pour la durée de la session en cours, ou à chaque nouvelle session.

Chapitre 23

Garder un système à jour

Comme expliqué précédemment, les logiciels malveillants se faufilent dans nos ordinateurs, entre autres, par l'intermédiaire de « failles de sécurité ». page 24

Des corrections pour ces erreurs de programmation (ou de conception) sont régulièrement mises à disposition, au fur et à mesure qu'elles sont identifiées. Une fois que ces corrections sont disponibles, il est particulièrement important de remplacer les anciennes versions des logiciels. En effet, les problèmes corrigés, qui pouvaient n'avoir auparavant été identifiés que par quelques spécialistes, sont ensuite connus et référencés publiquement… donc plus faciles à exploiter.

23.1 Garder Tails à jour

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 30 minutes à 1 heure, plus environ 30 minutes de téléchargement.

Un système live étant une collection indivisible de logiciels, exécutés à partir d'un DVD ou d'une clé USB, ou d'une carte SD, la seule solution praticable pour utiliser les dernières versions de ces logiciels est de s'assurer qu'on utilise bien la dernière version du système *live*. page 99

Au démarrage du système *live Tails*, le *Tails upgrader* apparaît pour nous prévenir lorsqu'une nouvelle version qui corrige des failles de sécurité est disponible.

Dans le cas où l'on utilise un DVD, il faut donc détruire celui contenant l'ancienne version et en graver un nouveau. Sauf si celui-ci est réinscriptible, auquel cas il suffira de l'effacer pour y graver la dernière version de Tails.

Pour une clé USB, ou une carte SD, et dans la mesure où l'on dispose d'une connexion Internet, on peut utiliser le *Tails upgrader* directement. Il suffit de cliquer sur *mettre à jour maintenant* et de suivre l'assistant tout au long du processus. Si une erreur se produit, ou s'il est nécessaire d'utiliser une autre méthode de mise à jour, l'assistant nous orientera vers la page de la documentation appropriée.

Celle-ci se trouve facilement à partir de la *Documentation de Tails* se trouvant sur le bureau. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Premier pas avec Tails* et cliquer sur la page *Mettre à jour une clé USB ou une carte SD Tails*.

23.2 Garder à jour un système chiffré

Une fois installé, un système chiffré doit être gardé à jour pour qu'on puisse continuer page 105

de lui faire confiance. Les sections qui suivent concernent le système *Debian*, mais les concepts s'appliquent dans les grandes lignes à quasiment tous les autres systèmes.

Le projet *Debian* publie, à peu près tous les deux ans, une version *stable*. Cela représente un énorme effort pour coordonner la compatibilité des différentes versions des logiciels, effectuer de nombreux tests et s'assurer qu'il n'y reste aucun défaut majeur.

23.3 Les mises à jour quotidiennes d'un système chiffré

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : une minute pour lancer la mise à jour, plus un temps variable pour les téléchargements et l'installation, pendant lequel on peut continuer à utiliser son ordinateur.

Tout l'intérêt d'une version *stable* de Debian est que par la suite, les logiciels qui la composent ne sont plus modifiés en profondeur : ne seront ajoutées que les améliorations de traduction, les corrections de problèmes liés à la sécurité ou empêchant d'utiliser normalement un programme.

Ces nouvelles versions peuvent donc être en général installées « les yeux fermés », elles ne devraient pas perturber les petites habitudes qu'on a prises.

Lorsqu'on a installé l'*environnement graphique de bureau*, le système vérifiera automatiquement, lorsqu'il sera connecté à Internet [1], la disponibilité de nouvelles versions dans les dépôts configurés.

page 122

Lorsque c'est le cas, une fenêtre ainsi qu'une icône apparaîtront dans l'espace de notification afin de proposer de procéder aux mises à jour.

Après avoir cliqué sur l'icône, le système nous demande d'entrer le mot de passe d'administration. Une fois cela fait, une fenêtre s'ouvre avec la liste des paquets qui peuvent être mis à jour. Ils sont normalement tous sélectionnés. Il suffit de cliquer sur le bouton *Installer les mises à jour* (ou en anglais *Install updates*) pour lancer la procédure.

23.4 Passage à une nouvelle version stable

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : une demie journée à une journée, dont un long temps de téléchargement pendant lequel on peut continuer à utiliser son ordinateur, et un long temps d'installation pendant lequel il vaut mieux ne plus l'utiliser.

Lorsqu'une nouvelle version *stable* de Debian est sortie, le projet veille à garder à jour la précédente version *stable* pendant une durée d'**un an**.

Il est donc nécessaire de profiter de cette période pour prendre le temps de mettre à jour son système vers cette nouvelle version. C'est un processus plus délicat que les mises à jour quotidiennes – pas nécessairement dans sa réalisation même, mais dans

1. On peut également effectuer ces petites mises à jour (certes un peu moins quotidiennement) sur un ordinateur qui n'aurait pas du tout accès au réseau. Le projet Debian sort régulièrement de nouvelles versions *mineures* (ou *point releases* en anglais) qui sont annoncées sur le site web du projet. Le projet propose alors des DVD contenant tous les paquets mis à jour ; un tel DVD peut par exemple s'appeler `debian-update-7.4.0-amd64-DVD-1.iso`. En ajoutant ce DVD aux *Sources de mise à jour*, il est ensuite possible d'utiliser le *Gestionnaire de mises à jour* sans pour autant que l'ordinateur soit connecté.

le fait qu'il est ensuite nécessaire de s'adapter aux changements qu'auront connus les logiciels que nous utilisons habituellement.

23.4.1 Passage de *Squeeze* à *Wheezy*

La procédure détaillée ici concerne la mise à jour de la version de Debian baptisée *Squeeze* ou 6.0, sortie en février 2011, à la version *Wheezy* ou 7.0, sortie en mai 2013.

Nous documenterons ici une procédure de mise à jour simplifiée qui a été testée sur des installations de Debian *Squeeze* avec un environnement graphique de bureau GNOME et des logiciels provenant uniquement des dépôts officiels de Debian.

Elle nécessite de disposer, pour la durée de la mise à jour, d'une connexion à Internet ou d'un DVD d'installation. page 106

Attention ! Cette procédure simplifiée a moins de chances de fonctionner lorsqu'on a bidouillé son système en ajoutant des sources de mises à jour non officielles.

Si c'est le cas, se référer aux notes de publication officielles du projet Debian [2], notamment la partie Mises à niveau depuis Debian 6.0 (Squeeze) [3] et la partie Problèmes à connaître pour Squeeze [4].

Mettre à jour sa Debian *Squeeze*

Avant tout, il est nécessaire de disposer d'une Debian *Squeeze* à jour. Sans cela, la mise à niveau risque fort de ne pas fonctionner. Au cas où ces mises à jour n'auraient pas été faites au quotidien, c'est le moment de rattraper le retard. S'il vous est proposé de redémarrer, suite à de nombreuses mises à jour, le faire avant de procéder à la suite des opérations. page précédente

S'assurer d'avoir assez d'espace libre sur le disque dur

Avant d'éviter toute mauvaise surprise, il faut avoir au moins 4 Go d'espace libre sur le disque dur qui contient le système.

On peut vérifier cela en ouvrant le *Poste de travail* à partir de menu *Raccourcis*. Ensuite, on effectue un clic-droit sur l'icône du *Système de fichiers* pour atteindre les *Propriétés*. Dans la fenêtre qui s'ouvre, l'information qui nous intéresse se trouve en bas à droite du graphique, avant l'étiquette *libre*.

Libérer de l'espace sur le disque si nécessaire S'il n'y a pas assez d'espace sur le disque dur, une piste est d'effacer d'anciennes mises à jour devenues obsolètes. Pour cela, ouvrir le *Gestionnaire de paquets Synaptic* à partir du menu *Système → Administration*. Dans le menu *Configuration* choisir *Préférences*, puis sélectionner l'onglet *Fichiers* et cliquer sur le bouton *Supprimer les paquets en cache*, puis fermer le *Gestionnaire de paquets Synaptic*.

Si cela ne suffit pas, il faudra supprimer certains de nos propres fichiers ou désinstaller des programmes.

Désactiver les autres dépôts Debian

La mise à jour n'est testée qu'avec les paquets officiellement fournis par Debian *Squeeze*. On va donc désactiver tous les autres dépôts Debian, y compris les dépôts *backports* et *volatile*. page 122

Pour cela, ouvrir les *Sources de mises à jour* à partir du menu *Système → Administration*. Vu qu'on va choisir à quels programmes on fait confiance, on doit entrer le

2. http://www.debian.org/releases/wheezy/i386/release-notes/index.fr.html
3. http://www.debian.org/releases/wheezy/i386/release-notes/ch-upgrading.fr.html
4. http://www.debian.org/releases/wheezy/i386/release-notes/ch-information.fr.html

mot de passe d'administration. Dans l'onglet *Third-party software*, décocher tous les dépôts listés, puis cliquer sur *Fermer*.

À ce moment là, si le logiciel demande s'il faut recharger les informations sur les paquets disponibles, lui confirmer en cliquant sur *Actualiser*.

Mettre à jour les dépôts Debian utilisés

Commençons par modifier les dépôts configurés afin d'utiliser ceux dédiés à la nouvelle version. Il n'est pas encore possible de faire cette opération à travers l'interface graphique, il faut donc ouvrir un *Terminal administrateur* et taper la commande suivante :

page 85

```
gedit /etc/apt/sources.list
```

L'éditeur de texte s'ouvre. Dans les menus, on sélectionne ensuite *Rechercher* → *Remplacer*. Dans la fenêtre qui s'ouvre, *Rechercher* « `squeeze` » pour le *Remplacer par* « `wheezy` ». Cliquer ensuite sur le bouton *Tout remplacer*, puis *Fermer* la fenêtre de recherche.

Si une installation ou une mise à jour a été faite auparavant en utilisant un CD ou un DVD, c'est une bonne idée de chercher les lignes qui commencent par « `deb cdrom:` » pour les supprimer.

On peut ensuite quitter l'éditeur en enregistrant les modifications.

Dans une version précédente de ce *guide* on conseillait la création d'un fichier qui n'est plus nécessaire mais pourrait nuire à la mise à jour. Le supprimer grâce à la commande :

```
rm /etc/apt/preferences
```

Si jamais l'ordinateur nous répond `rm: impossible de supprimer « /etc/apt/preferences »: Aucun fichier ou dossier de ce type`, c'est simplement que le fichier est déjà absent.

Nous avons modifié la liste des dépôts ; il faut donc maintenant télécharger la liste des paquets qui y sont disponibles, avant de pouvoir les installer ; pour cela, toujours le *Terminal administrateur* qu'on gardera ouvert, taper la commande :

```
apt-get update
```

Ajouter le DVD d'installation à la liste des dépôts Debian

La méthode la plus simple pour effectuer les mises à jour consiste à laisser Debian les télécharger directement depuis Internet. Au cas où l'ordinateur n'est pas relié à Internet, ou si la connexion est de mauvaise qualité, on peut demander au système d'utiliser un DVD d'installation de Debian comme dépôt de paquets. Si ce n'est pas le cas, passer directement à l'étape suivante.

Pour cela, reprendre le *Terminal administrateur* et taper la commande suivante :

```
apt-cdrom add
```

Il faut ensuite insérer le DVD, et appuyer sur la touche *Entrée*. On peut garder le terminal ouvert, il devrait nous resservir sous peu.

Désactiver l'économiseur d'écran

Lors de la mise à jour, l'économiseur d'écran peut se bloquer, et laisser l'écran verrouillé. Il est donc prudent de le désactiver pour le temps de la mise à jour.

Pour cela, ouvrir *Économiseur d'écran* à partir du menu *Système* → *Préférences*. Dans la fenêtre qui s'ouvre, décocher *Activer l'économiseur d'écran quand l'ordinateur est inactif*. Fermer cette fenêtre.

Lancer la mise à jour proprement dite

La mise à jour se fait en plusieurs étapes que l'on pilotera à l'aide du *Terminal administrateur*.

Notre première commande dit au gestionnaire de paquets, d'une part, que nous préférons qu'il nous pose le moins possible de questions concernant les détails de la mise à jour ; d'autre part, que ces questions doivent être posées dans une interface graphique ; enfin, qu'on ne veut pas voir l'historique des changements :

```
export DEBIAN_PRIORITY=critical DEBIAN_FRONTEND=gnome
↪    APT_LISTCHANGES_FRONTEND=none
```

Notre deuxième commande effectue la première partie de la mise à jour du système :

```
apt-get upgrade
```

Assez rapidement, le terminal affiche `Souhaitez-vous continuer [O/n]?` Après avoir confirmé en appuyant sur *Entrée*, on peut voir apparaître une première série de fenêtres nous demandant comment gérer certains changements.

Lorsqu'on ne cherche pas à sortir des choix de Debian, cliquer sur *Suivant* (*Forward* lorsque les choix sont en anglais) à chaque fois est suffisant.

Au bout d'un moment, un certain nombre de paquets ont déjà été mis à jour, et le terminal devrait revenir à l'invite de commande.

La quatrième commande terminera la mise à jour du système :

```
apt-get dist-upgrade
```

Encore une confirmation, toujours avec *Entrée*, et c'est parti. On peut voir apparaître une nouvelle série de fenêtres. Sauf à vouloir sortir des propositions de Debian, on cliquera sur *Suivant* (parfois écrit en anglais : *Forward*).

À cette étape de la mise à jour, il peut arriver que le bureau GNOME affiche divers messages d'erreurs. Ce n'est pas particulièrement inquiétant, dans la mesure où l'on est en train de réinstaller de nombreux composants du système. Ces problèmes devraient se résoudre d'eux-mêmes une fois le processus terminé.

Quelques évolutions du système plus tard, le terminal nous invite une nouvelle fois à lui indiquer des commandes.

Quand l'invite réapparaît, on peut alors saisir une dernière commande, pour libérer de l'espace disque :

```
apt-get clean
```

Premier redémarrage

Le moment est maintenant venu de redémarrer le système.

Dans notre terminal administrateur, taper la commande `reboot` et appuyer sur la toucher *Entrée*.

Réactiver les dépôts Debian supplémentaires

On peut maintenant souffler. Le plus gros est fait. Il reste toutefois encore quelques petits ajustements…

Si l'on a désactivé des dépôts non officiels avant la mise à jour, c'est le moment de vérifier qu'on en a toujours besoin avec la nouvelle version de Debian. Si oui, page 122 les réactiver. On peut également réactiver l'économiseur d'écran si on l'a désactivé auparavant.

Prendre en main le nouveau système.

Après le redémarrage, on se retrouve dans la nouvelle version de Debian. Si l'on a un ordinateur suffisamment puissant, on se retrouve avec un nouvel environnement de bureau appelé *Gnome Shell*, auquel on mettra peut-être un peu de temps à s'habituer. Pour une introduction à cette nouvelle interface, on pourra consulter l'aide en cliquant sur *Activités* en haut à gauche de l'écran puis sur *Applications*, *Accessoires* et enfin *Aide*.

S'assurer que le nouveau système fonctionne correctement

Il peut être utile de s'assurer que les actions et les commandes les plus courantes sont fonctionnelles. Le cas échéant, il pourrait être nécessaire de diagnostiquer et de résoudre les problèmes. Il vaut certainement mieux le faire dès la prise de contact avec le nouveau système, afin de pouvoir repartir pour deux ans avec un système fonctionnel. Les problèmes les plus courants sont souvent décrits, avec les astuces page 114 pour les résoudre, dans diverses documentations sur Debian et GNU/Linux.

Rappelons également qu'il existe des notes de publication officielles du projet Debian [5].

5. http://www.debian.org/releases/wheezy/i386/release-notes/index.fr.html

Chapitre 24

Nettoyer les métadonnées d'un document

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : quelques minutes.

L'objectif de cet outil est d'effacer les métadonnées présentes dans un document avant sa publication. Ces métadonnées ne sont pas les mêmes dans tous les formats de documents : certaines sont plus difficiles (voire impossibles) à nettoyer que d'autres. Cependant, la plupart des formats utilisés pour échanger des documents terminés, que ce soient des textes, des images, du son ou de la vidéo, sont « nettoyables ».

page 21

L'outil qu'on va utiliser ici est le *Metadata Anonymisation Toolkit* (*MAT*) qui permet de nettoyer aisément de nombreux formats de fichiers.

Attention ! Nettoyer les métadonnées n'anonymise pas le contenu des fichiers, et n'enlève pas les éventuels marquages [1] qui seraient inclus dans le contenu lui-même.

24.1 Installer les logiciels nécessaires

Sous Tails, le *Metadata Anonymisation Toolkit* est déjà installé. Sur un système où le paquet `mat` n'est pas encore présent, il faut l'installer.

page 120

24.2 Ouvrir le *Metadata Anonymisation Toolkit*

Aller dans *Applications* → *Outils Système* → *MAT*.

24.3 Ajouter des fichiers à nettoyer

Ajouter le fichier à nettoyer avec *File* → *Ajouter* puis après l'avoir selectionné cliquer sur *Valider*. Il est également possible de le glisser directement dans *Metadata Anonymisation Toolkit*. On pourra également ajouter plusieurs fichiers et les nettoyer en même temps. Si le logiciel sait nettoyer le fichier sélectionné, il l'ajoute à la liste de fichiers à nettoyer. Passer alors directement à *Scour* (nettoyer les fichiers) ci-dessous.

S'il affiche un message « Fichier(s) non supporté(s) », il faut convertir le document dans un format de fichiers supporté par le *MAT* avec une autre application avant de

1. Voir à ce sujet Wikipédia, 2014, *Tatouage numérique* [https://fr.wikipedia.org/wiki/Tatouage_numérique] et Wikipédia, 2014, *Stéganographie* [https://fr.wikipedia.org/wiki/Stéganographie].

pouvoir le nettoyer. La liste des formats supportés est disponible à partir de *Help* → *Supported Formats*.

Souvent, il suffit de l'exporter dans un format d'échange de fichiers commun. Ainsi, le *MAT* ne peut pas nettoyer le fichier *XCF* du programme de manipulation d'images GIMP, mais il sait nettoyer les images exportées en *JPEG* ou *PNG*.

24.4 Nettoyer les fichiers

Une fois le fichier ajouté, le nettoyer avec *Process* → *Scour*. On peut alors fermer *Metadata Anonymisation Toolkit*.

page 129

Pour supprimer les fichiers contenant les métadonnées, voir l'outil expliquant comment effacer des fichiers et leur contenu.

Qui parle ?

D'où vient cet ouvrage ? Qui parle, en ses lignes ?

Nous pourrions nous contenter de dire qu'il nous semble parfaitement inintéressant de chercher des réponses à de telles interrogations ; que nous laissons aux flics, spécialistes de la question, le privilège de s'y consacrer ; que nous avons mieux à faire.

Le fait que telle ou telle personne couche des mots sur le papier n'est pas, croyons-nous, particulièrement déterminant dans le contenu d'un texte, dans son existence même.

Nous croyons plutôt qu'il s'écrit lorsque des désirs s'entremêlent, lorsque des nécessités se confrontent, lorsque des questions appellent des réponses. Des façons de se rapporter à ce qui nous entoure se rencontrent, se partagent, se transforment alors. Elles se lient, et des manières *communes* de s'y rapporter se construisent, qui interagissent avec d'autres : cela va des conflits aux complicités, en passant par l'alliance et le clin d'œil entendu ; sont alors en jeu sensibilités, critères éthiques, calculs stratégiques…

Bien plus que la « pensée » de X ou Y, un livre exprime l'état de ces interactions, à un certain moment.

*
* *

Deux caractéristiques de cet ouvrage nous obligent néanmoins à faire face, sous certains angles, aux interrogations relatives à sa provenance. Cet ouvrage prétend d'une part transmettre des savoirs et savoirs-faire techniques, réservés d'ordinaire à de rares spécialistes. D'autre part, la justesse des indications fournies peut avoir de larges implications sur la sérénité des personnes qui les mettraient en œuvre. Les petites erreurs qui nous auront échappé peuvent donc avoir de graves conséquences.

Il importe donc de dire quelques mots sur les bouches qui ont prêté leurs voix à ce guide. Mettre au clair l'étendue de nos savoirs(-faire) — et leurs limites — permet de trouver un rapport d'apprentissage plus adéquat à cet écrit, mais aussi de décider du niveau de confiance *technique* qu'il mérite. Disons donc que, collectivement :

- les questions brassées par ce guide nous traversent, techniquement et politiquement, depuis une dizaine d'années ;
- nous connaissons très bien le fonctionnement des systèmes d'exploitation, et particulièrement celui de Debian GNU/Linux ;
- nous avons des bases solides en cryptographie, mais sommes très loin de pouvoir prétendre à une quelconque expertise en la matière.

Et pour finir, affirmons une dernière fois que la parole portée par cet ouvrage, comme toute parole de *guide*, se doit d'être prise avec des pincettes d'autant plus longues que ses implications sont importantes.

Index

A
algorithme, 38
application, 15
architecture, 11
archivage, 77
argument, 86
authenticité, 41

B
backdoor, *voir* porte dérobée
backports, 123
bibliothèque, 15
binaire, 11
BIOS, 13, 61
boot, *voir* démarrage
bug, 20

C
cache, 33
carte-mère, 10
chemin d'un fichier, 86
cheval de Troie, 24
chiffrement, 37
    chiffrer un système, 105
    chiffrer une clé, 137
clé de chiffrement, 38, 40
code source, 29
cold boot attack, 19, 40, 61
collision, 42
confidentialité, 37
CPU, *voir* processeur
cryptanalyse, 37
cryptographie, 37
    cryptographie asymétrique, 43
    cryptographie symétrique, 43
cryptologie, 40

D
Debian, 15, 105
Déjà Dup, 145
démarrage, 93
disque dur, 12, 32
distribution, 30
dépôt de paquets, 122

E
écrasement des données, 32
effacement, 32
électricité, 14
empreinte, *voir* somme de contrôle
en-tête LUKS, 40
enregistreur de frappe, 26
espace d'échange, *voir* mémoire virtuelle
ext2, ext3, 16

F
FAT32, 16
fonction de hachage, 42
force brute, 63
format de fichiers, 17
formatage, 16, 34

G
GNU/Linux, 15, 30
GnuPG, 39

H
hachage, 42
HADOPI, 25
hibernation, 20
historique, 20

I
imprimante, 27
installation d'un logiciel, 115
installation d'un système, 105
installeur, 105
intégrité, 41

J
journal, 20
journalisation, 33

K
keylogger, *voir* enregistreur de frappe

L
library, *voir* bibliothèque
licence libre, 30
licence propriétaire, 29
ligne de commande, 85
liste blanche, 53
liste noire, 53
log, *voir* journaux
logiciel, 14
    logiciel espion, 24
    logiciel libre, 29, 30
    logiciel malveillant, 24, 62
    logiciel open source, 30
    logiciel portable, 34
    logiciel propriétaire, 29
logiciel malveillant, 23
LOPPSI 2, 23
LUKS, 40, 137

M
malware, *voir* logiciel malveillant
mémoire *flash*, 13, 32, 128
mémoire morte, *voir* mémoire persistante
mémoire persistante, 12
mémoire virtuelle, 17, 19
mémoire vive, 11, 19
micrologiciel, 13, 61
mise à jour, 169
mot de passe, 31
méta-données, 21

N
noyau, 15
NTFS, 16
numérisation, 10

O
ondes, 14
open source, 30
option, 86
OS, *voir* système d'exploitation

P
paquet Debian, 115
partition, 16
phrase de passe, 37, 91
pilote, 15
politique de sécurité, 51
porte dérobée, 29
pourriel, 23, 24
processeur, 10
programme, 14
périphérique, 13

R
RAM, *voir* mémoire vive
rootkit, 24

S
sauvegarde, 143
sauvegardes automatiques, 21
`secure-delete`, 127
`sfill`, 135
`shred`, 131, 134
signature numérique, 44
signature stéganographique, 27
somme de contrôle, 41, 153
spam, *voir* pourriel
spyware, *voir* logiciel espion
SSD, *voir* mémoire *flash*
stéganographie, 27
swap, 17, 19
Synaptic, 116, 120
syntaxe, 86
système de fichiers, 16, 33
système d'exploitation, 15
    système hôte, 70
    système invité, 70
    système *live*, 34, 68, 99

T
terminal, 85
traces, 19
transistor, 10
Transmission, 100
TrueCrypt, 30

U
UEFI, 13, 61
upgrade, *voir* mise à jour
USB, 13

V
veille, 20
VirtualBox, 70, 156
virtualisation, 69
virus, 24

W
watermarking, *voir* signature stéganographique
Windows, 68, 157
wipe, *voir* écrasement des données

Photo page 10 de Darkone, licence CC BY-SA 2.5, trouvée sur :
https://secure.wikimedia.org/wikipedia/fr/wiki/Fichier:ASRock_K7VT4A_Pro_Mainboard.jpg.

Photo page 10, domaine public, trouvée sur :
https://secure.wikimedia.org/wikipedia/fr/wiki/Fichier:Pentium-60-back.jpg

Photo page 12, domaine public, trouvée sur :
https://secure.wikimedia.org/wikipedia/fr/wiki/Fichier:DDR_RAM-3.jpg

Photo page 12, domaine public, trouvée sur :
https://secure.wikimedia.org/wikipedia/fr/wiki/Fichier:Hdd-wscsi.jpg

Photo page 13 de Zac Luzader Codeczero, licence CC BY 3.0, trouvée sur :
https://secure.wikimedia.org/wikipedia/fr/wiki/Fichier:AT_Motherboard_RTC_and_BIOS.jpg.

guide d'autodéfense numérique

tome 1

hors connexions

[…] *nous n'avons pas envie d'être contrôlables par quelque « Big Brother » que ce soit. Qu'il existe déjà ou que l'on anticipe son émergence, le mieux est sans doute de faire en sorte qu'il ne puisse pas utiliser, contre nous, tous ces merveilleux outils que nous offrent — ou que lui offrent — les technologies numériques.* […]

Même si l'on choisit de ne pas les utiliser directement, d'autres le font pour nous. Alors, autant essayer de comprendre ce que ça implique.

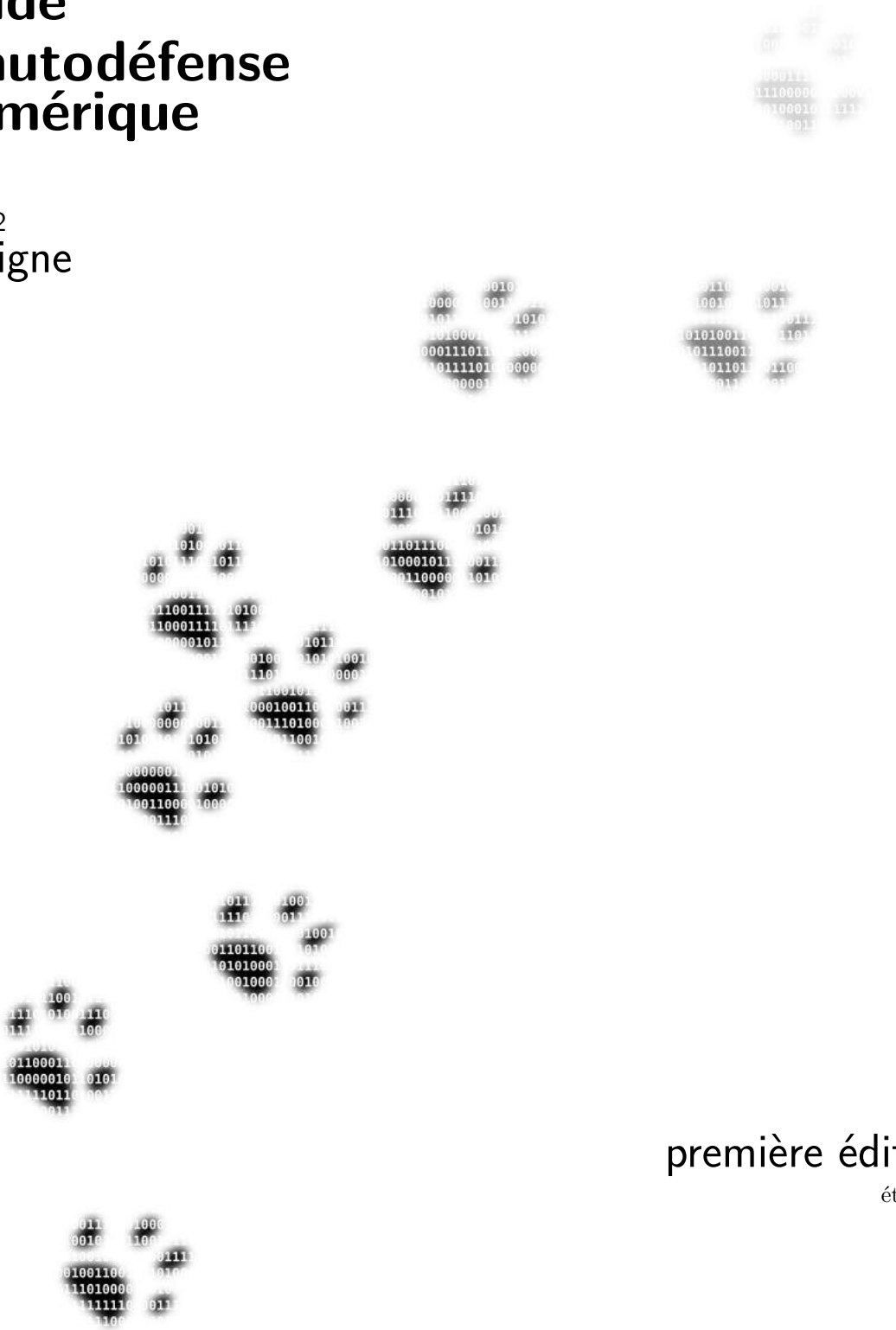
Face à ces constats, la seule voie praticable semble être de devenir capables d'imaginer et de mettre en place des politiques de sécurité adéquates.
Tout l'enjeu de ce guide est de fournir cartes, sextant et boussole à quiconque veut cheminer sur cette route.

Ce premier tome se concentre sur l'utilisation d'un ordinateur « hors connexion » — on pourrait aussi bien dire préalablement à toute connexion *: les connaissances générales qu'il apporte valent* que l'ordinateur soit connecté ou non *à un réseau.*

Un livre à lire, relire, pratiquer, en solitaire ou à plusieurs, à faire découvrir et à partager… ou comment affiner l'art de la navigation dans les eaux troubles du monde numérique.

guide d'autodéfense numérique

tome 2
en ligne



première édition
été 2014

ouvrage collectif

Guide d'autodéfense numérique
Tome 2 : En ligne
première édition

Ouvrage collectif
guide@boum.org

Empreinte OpenPGP :
D487 4FA4 F6B6 88DC 0913
C9FD 326F 9F67 250B 0939

été 2014

Copyleft : cette œuvre est libre, vous pouvez la copier, la diffuser et la modifier selon les termes de la *Licence Art Libre* — http://www.artlibre.org/

Préface

Le premier tome de ce *Guide* s'est attaché à donner quelques bases sur le fonctionnement des ordinateurs hors ligne, afin de mettre en lumière ce qu'ils peuvent dévoiler sur leurs utilisateurs, puis de proposer quelques cas concrets de leur usages et enfin des outils associés aux problématiques soulevées. Comme annoncé, ce second volet s'intéressera donc à l'utilisation des ordinateurs en ligne. Vaste programme… car si une plongée dans les arcanes de ces machines pourtant familières s'était déjà avérée un brin complexe, qu'en sera-t-il maintenant qu'on se propose de connecter les ordinateurs entre eux ? Rappelons dès à présent qu'un ordinateur connecté est avant tout un ordinateur ; la (re)lecture du premier tome est donc un prérequis essentiel pour appréhender toutes les facettes de la sécurité *en ligne*.

Internet, pourtant, nous est également devenu familier. Consulter ses mails, télécharger des fichiers, obtenir des informations en ligne sont aujourd'hui pour beaucoup d'entre nous des gestes quotidiens. Chacun pourrait dire que d'une certaine manière, il *sait ce que c'est* qu'Internet. Admettons plutôt que tout le monde, ou presque, est capable de s'en servir pour quelques usages communs.

Notre propos dans ce second tome, pour autant, ne sera pas de définir dans les moindres détails ce qu'est Internet. Tout au plus fournira-t-on quelques éléments de compréhension suffisants pour permettre au lecteur d'y naviguer — ambiguïté du terme, qui renvoie autant à la « navigation sur le web » qu'à la possibilité de s'orienter dans un espace complexe à l'aide d'outils adaptés. Ou le retour du sextant et de la boussole…

Commençons par le début. Internet est un réseau. Ou plutôt, un ensemble de réseaux connectés entre eux qui, à partir d'une obscure application à visée militaire, s'est étendu au fil de dizaines d'années au monde entier. Réseau qui a vu se multiplier les applications, les usages et les usagers, les technologies et les techniques de contrôle.

Certains ont pu disserter à l'infini sur le « nouvel âge » qui s'ouvrait, les supposées possibilités d'horizontalité et de transparence dans la diffusion de l'information et des ressources, ou dans l'organisation collective, auxquelles a pu ouvrir cette nouvelle technologie — y compris dans l'appui qu'il pouvait offrir pour les luttes politiques. Cependant, comme il semble évident que les pouvoirs n'aiment pas ce qui peut leur échapper, même partiellement, il s'est développé, en même temps que l'expansion des usages, une expansion des techniques de contrôle, de surveillance et de répression, dont les conséquences se font de plus en plus sentir.

Au cours de l'année 2011, pour la première fois, des gouvernements ont organisé la déconnexion de la quasi-totalité de leurs habitants vis-à-vis du réseau mondial. L'Égypte, comme l'Iran, puisque c'est d'eux qu'il s'agit, ont estimé que pour mieux

contenir les révoltes qui prenaient place sur leurs sols, ils avaient tout intérêt à limiter au maximum les possibilités de communication par le réseau — ce qui ne les a pas empêchés, dans le même mouvement, de chercher à organiser la surveillance et le pistage sur le réseau. L'Iran fut ainsi capable de mettre en place un système d'analyse de trafic demandant des ressources importantes pour surveiller les opposants, connus ou non, établir une cartographie de leurs relations et plus tard confondre et condamner les révoltés qui utilisaient le réseau pour s'organiser.

Autre exemple, depuis la mise en place d'une version chinoise de Google [1] en 2006, et jusqu'en 2010, l'entreprise acceptait la politique du gouvernement chinois de filtrage des résultats de recherche.

Des méthodes similaires ont aussi cours dans des pays dits démocratiques. Ainsi, à la fin de l'été 2011, après plusieurs journées d'émeutes à Londres, deux jeunes anglais ont été condamnés [2] à 4 ans de prison pour avoir appelé sur Facebook à des rassemblements dans leurs quartiers – et ce, alors même que leurs « appels » n'ont pas été suivis.

De même, les révélations d'Edward Snowden [3] sur l'état de la surveillance électronique mise en place par la NSA [4] à l'échelle mondiale ont rendu crédibles les hypothèses parmi les plus pessimistes. À partir de là, il apparaît indispensable de prendre conscience que l'utilisation d'Internet, tout comme celle de l'informatique en général, est tout sauf anodine. Elle nous expose à la surveillance, et à la répression qui peut lui succéder : c'est l'objet principal de ce second tome que de permettre à tout un chacun de comprendre quels sont les risques et les limites associés à l'utilisation d'Internet. Mais il s'agit aussi de se donner les moyens de faire des choix éclairés quant à nos usages de l'Internet. Des choix qui peuvent permettre de compliquer la tâche des surveillants, de contourner des dispositifs de censure, voire de mettre en place des outils, des infrastructures, de manière autonome. Une première amorce pour reprendre le contrôle de technologies qui semblent parfois vouées à nous échapper – ambition qui dépasse cependant largement les objectif de ce guide.

Nous voici donc repartis pour un nouveau voyage dans les eaux troubles du monde numérique. Notre traversée se fera en trois parties, une première expliquant le contexte, les notions de base, permettant une compréhension générale, une seconde partie traitant de cas d'usage typiques, et enfin une troisième décrivant précisément les outils nécessaires à la mise en œuvre de politiques de sécurité abordées dans la seconde partie ainsi que leurs usages.

1. Wikipédia, 2014, *Google China* [https://fr.wikipedia.org/wiki/Google_China].
2. France Soir, 2011, *émeutes à Londres : Deux jeunes condamnés à quatre ans de prison* [http://www.francesoir.fr/actualite/international/emeutes-londres-deux-jeunes-condamnes-quatre-ans-prison-128302.html].
3. Wikipédia, 2014, *Edward Snowden* [https://fr.wikipedia.org/wiki/Edward_Snowden]
4. *National Security Agency*, agence dépendant du département de la Défense des États-Unis, chargée de la collecte et de l'analyse des données étrangères et de la protection des données étatsuniennes.

Tome 2

En ligne

Sommaire

Préface iii

Sommaire 1

I Comprendre 5

1 Bases sur les réseaux 9
1.1 Des ordinateurs branchés entre eux . . . 9
1.2 Protocoles de communication . . . 11
1.3 Les réseaux locaux . . . 13
1.4 Internet : des réseaux interconnectés . . . 16
1.5 Des applications variées . . . 19
1.6 Des clients, des serveurs . . . 21

2 Traces sur toute la ligne 25
2.1 Sur l'ordinateur client . . . 25
2.2 Sur la « box » : l'adresse matérielle de la carte réseau . . . 28
2.3 Sur les routeurs : les en-têtes de paquets . . . 28
2.4 Sur le serveur . . . 29
2.5 Les traces qu'on laisse soi-même . . . 31

3 Surveillance et contrôle des communications 33
3.1 Qui veut récupérer les données ? . . . 33
3.2 Journaux et rétention de données . . . 36
3.3 Écoutes de masse . . . 40
3.4 Attaques ciblées . . . 41
3.5 En conclusion . . . 47

4 Web 2.0 49
4.1 Des « applications Internet riches »... . . . 49
4.2 ...et des clients devenus bénévoles . . . 50
4.3 Centralisation des données . . . 50
4.4 Mainmise sur les programmes . . . 51
4.5 De la centralisation à l'auto-hébergement décentralisé . . . 52

5 Identités contextuelles 53
5.1 Définitions . . . 53
5.2 De l'identité contextuelle à l'identité civile . . . 54
5.3 La compartimentation . . . 56

5.4 Les médias sociaux : centralisation de fonctions et identité unique . . 56

6 Cacher le contenu des communications : la cryptographie asymétrique 59
6.1 Limites du chiffrement symétrique 59
6.2 Une solution : la cryptographie asymétrique 59
6.3 Signature numérique 62
6.4 Vérifier l'authenticité de la clé publique 63
6.5 Confidentialité persistante 67
6.6 Résumé et limites 68

7 Cacher les parties prenantes de la communication : le routage en oignon 69
7.1 Présentation du routage en oignon 69
7.2 Participer au réseau Tor 72
7.3 Quelques limites de Tor 72

II Choisir des réponses adaptées 77

8 Consulter des sites web 79
8.1 Contexte 79
8.2 Évaluer les risques 79
8.3 Définir une politique de sécurité 80
8.4 Choisir parmi les outils disponibles 82
8.5 Naviguer sur des sites web avec le Tor Browser Bundle 84
8.6 Naviguer sur des sites web avec Tails 84

9 Publier un document 87
9.1 Contexte 87
9.2 Évaluer les risques 87
9.3 Définir une politique de sécurité 87
9.4 Contact public 89

10 Échanger des messages 91
10.1 Contexte 91
10.2 Évaluer les risques 91
10.3 Deux problématiques 92
10.4 Webmail ou client mail ? 93
10.5 Webmail 93
10.6 Client mail 94
10.7 Échanger des emails en cachant son identité 95
10.8 Echanger des emails confidentiels (et authentifiés) 97

11 Dialoguer 101
11.1 Contexte 101
11.2 Évaluer les risques 101
11.3 Définir une politique de sécurité 101
11.4 Les limites 103

III Outils 107

12 Installer et configurer le Tor Browser Bundle 111

12.1 Télécharger et vérifier le *Tor Browser Bundle* . . . 112
12.2 Décompresser le Tor Browser Bundle . . . 113
12.3 Lancer le Tor Browser Bundle . . . 113

13 Naviguer sur le web avec Tor **115**
13.1 Lancer le navigateur . . . 115
13.2 Quelques remarques sur la navigation . . . 115

14 Choisir un hébergement web **117**
14.1 Quelques critères de choix . . . 117
14.2 Type de contenu . . . 118
14.3 En pratique . . . 119

15 Ajouter un certificat électronique à son navigateur **121**
15.1 Vérifier un certificat ou une autorité de certification . . . 121
15.2 Ajouter un certificat . . . 122
15.3 Ajouter une autorité de certification . . . 124
15.4 Trouver l'empreinte d'un certificat déjà installé . . . 124

16 Utiliser un clavier virtuel dans Tails **125**
16.1 Utiliser un clavier virtuel dans Tails . . . 125

17 Utiliser le client mail Claws Mail **127**
17.1 Installer le client mail Claws Mail . . . 127
17.2 Lancer Claws Mail . . . 127
17.3 Configurer un compte email . . . 127
17.4 Configuration avancée de Claws Mail . . . 128

18 Utiliser OpenPGP **131**
18.1 Importer une clé OpenPGP . . . 131
18.2 Vérifier l'authenticité d'une clé publique . . . 132
18.3 Signer une clé . . . 133
18.4 Créer et maintenir une paire de clés . . . 134
18.5 Exporter une clé publique OpenPGP . . . 137
18.6 Utiliser la crytographie asymétrique pour chiffrer ses emails . . . 138
18.7 Déchiffrer des emails . . . 139
18.8 Vérifier une signature numérique OpenPGP . . . 139
18.9 Signer des emails . . . 141
18.10 Signer des données . . . 141
18.11 Révoquer une paire de clés . . . 142

19 Utiliser la messagerie instantanée avec OTR **145**
19.1 Installer le client de messagerie instantanée Pidgin . . . 145
19.2 Lancer Pidgin . . . 145
19.3 Configurer un compte de messagerie . . . 145
19.4 Créer un compte de messagerie instantanée . . . 146
19.5 Chiffrer la connexion au serveur XMPP . . . 146
19.6 Activer le plugin *Off-the-Record* . . . 146
19.7 Mettre en place une conversation privée . . . 147

20 Gérer des mots de passe **149**
20.1 Choisir une bonne phrase de passe . . . 149
20.2 Utiliser un gestionnaire de mots de passe . . . 149

Index **153**

Première partie

Comprendre

Dans le premier tome du guide d'autodéfense numérique, nous avons commencé par expliquer dans quelle mesure l'utilisation d'ordinateurs pouvait constituer une menace pour nos intimités dans le monde numérique, notamment par rapport aux données qu'on leur confie. Ce tome ambitionne de faire de même dans le cas où ces ordinateurs sont connectés à Internet.

Octobre 2010, Paris

Ce matin, Alice arrive en avance au travail. Elle est employée à La Reboute, une entreprise de vente de vêtements par correspondance, située au dernier étage d'un immeuble rue Jaurès : « pfiou, 18 étages, vivement que cet ascenseur soit réparé ! ». Elle s'installe à son bureau, se penche et appuie sur le bouton d'allumage de l'ordinateur.

Sur l'écran de l'ordinateur, une petite fenêtre vient d'apparaître. « Connexion réseau établie ». Avant de se mettre au boulot, elle veut regarder ses emails. Alice clique sur l'icône du navigateur web, provoquant l'ouverture d'une fenêtre qui reste vierge quelques millisecondes, avant de faire apparaître la page d'accueil de Google. Tout en appréciant mentalement la page d'accueil « spéciale Halloween » de Google, Alice déplace le pointeur de sa souris et clique sur le lien Connexion. *Une fois la page chargée, elle y rentre son nom d'utilisatrice et son mot de passe, puis clique sur Gmail. Quelque part dans une obscure salle bondée d'ordinateurs, un disque dur grésille. Quelques secondes après avoir ouvert son navigateur web, Alice commence à parcourir sa boîte mail. Alors qu'elle consulte un email reçu du site « leboncoin.fr », son regard est attiré par le lien qui vient de s'afficher dans la colonne de droite : « Tiens, quelqu'un vend le même modèle d'appareil photo que celui que je cherche, juste au coin de la rue... je devrais peut-être y faire un saut. »*

– *« Ah ben t'es là ? »*

La voix dans le dos d'Alice la fait légèrement sursauter. C'est Benoît, un collègue.

– *« Ben oui, je me suis levée un peu plus tôt que d'habitude, alors j'ai pris le RER de 7h27 au lieu de 7h43. Je regarde vite fait mes emails avant de*

m'y mettre. J'attends la confirmation d'une réservation de billet pour les Baléares cet hiver.
– *Vacances au soleil, j'vois le genre... Et t'en as pour longtemps ? »*

Benoît a l'air pressé.

– *« Euh... non non, j'avais presque fini. Pourquoi ?*
– *Ben, si ça te dérange pas, je t'emprunterais bien ton poste 2 minutes... Le mien est planté depuis hier, j'attend que la nouvelle responsable informatique, arrive pour régler ça ».*

Aussitôt assis, Benoît clique nerveusement sur la barre d'adresse du navigateur web, et rentre directement l'adresse du blog sur lequel sont régulièrement publiées des informations sur les personnages politiques de son arrondissement. Il n'aime pas passer par Google pour ses recherches, alors il l'a apprise par cœur. Sait-on jamais, ça pourrait éviter les mouchards. Ouvrant un deuxième onglet, il entre également l'adresse de no-log, sa boîte mail, et s'y connecte. Nickel, il est là ! Le document concernant les comptes bancaires en Suisse du Maire de son arrondissement, M. Alavoine ! Benoît télécharge aussitôt le document et l'ouvre dans l'éditeur de texte. Il le parcourt rapidement, et supprime quelques informations qu'il vaut mieux ne pas laisser. Après avoir entré son identifiant et son mot de passe pour se connecter au blog, Benoît copie-colle le contenu du document depuis sa boîte mail, et clique sur Envoyer. *« Espérons que cela inspire d'autres personnes ! »*

Satisfait d'avoir pu enfin envoyer son document, Benoît se relève aussitôt et rend sa place à Alice.

– *« On va se prendre un café ? »*

Novembre 2010. Siège social de La Reboute

En arrivant au bureau, Samuel Coustant, PDG de La Reboute, commence par éplucher le courrier reçu en buvant son café. Une convocation au commissariat. Pour une fois, il y a autre chose que des factures ! Sans doute une erreur ou une enquête de voisinage ?

Samuel ne pense pas avoir quoi que ce soit à se reprocher, alors inutile de s'inquiéter. Il se rend donc au commissariat le jour de sa convocation.

– *« M. Coustant ? Bonjour, nous voudrions vous poser quelques questions concernant une plainte pour diffamation... »*

Plus tard le même jour. Bureau d'Alice

– *« Allo ressources humaines de La Reboute, Alice j'écoute.*
– *Bonjour, M. Coustant à l'appareil. Écoutez, je viens de passer 2 heures au poste de police. J'ai été interrogé quant à des documents bancaires publiés sur Internet et concernant un certain M.Alavoine, Maire du 10ème, dont j'ignorait l'existence jusqu'alors. En plus de ça, lors de mon audition, ils m'ont présenté un papier les autorisant à faire une perquisition aux bureaux rue Jaurès.*
– *Quelle histoire ! Mais quel rapport avec nos bureaux ?*
– *Et bien c'est également pour ça que je vous appelle. Ils affirment qu'ils ont toutes les preuves comme quoi ces documents ont été publiés depuis*

vos bureaux. Je leur ai dit que ce n'était pas moi, que je ne voyais pas de quoi ils parlaient. Ils ont fait des recherches, contacté je ne sais qui. Mais ils disent qu'une enquête a été ouverte, et qu'elle ira jusqu'au bout. Qu'ils retrouveront les responsables. Autant vous dire que je ne suis pas franchement rassuré. J'espère bien que vous n'y êtes pour rien et qu'il s'agit d'un regrettable erreur.

– *Honnêtement, j'en suis la première étonnée, je ne vois absolument pas ce que j'aurais à voir là-dedans, ni ce dont il s'agit.*
– *J'espère bien… Enfin bref, c'est à la police de faire son travail désormais. Je vous rappellerai si j'ai des nouvelles de leur part.*
– *D'accord, je ferais de même s'ils appellent ici.*
– *Au revoir. »*

Alice repose le combiné, hébétée. Se gratte la tête. Mais qu'est-ce donc que cette histoire de documents bancaires ? Qui aurait pu faire ça ?

Commissariat central de Paris, quelques semaines plus tard.

– *« Commissaire Mathias ?*
– *Lui-même.*
– *Officier Nerret à l'appareil. Je vous appelle à propos de l'affaire Alavoine. On a eu un fax des collègues de la technique et scientifique qui ont les ordinateurs saisis entre les mains. Et on a du neuf.*
– *Allez-y, Nerret. Je vous écoute.*
– *Apparement, les collègues ont fini par retrouver le document sur un des ordinateurs. Il a été téléchargé depuis le navigateur, et modifié. Il y aurait eu une connexion à une boîte mail dont l'adresse correspond à une certaine Alice, chez Gmail, ainsi qu'une autre adresse email, chez no-log cette fois-ci, peu de temps avant la publication des documents incriminés.*
– *Ah, très bien. Mais vous ne comptez pas prendre un annuaire et interroger toutes les Alice de La Reboute quand même ?*
– *Non, on va d'abord la retrouver puis utiliser l'annuaire !*
– *Quel humour officier !*
– *On va demander à Gmail ainsi qu'à no-log les informations sur ces adresses email. À partir de là on pourra sans doute mettre la main sur les personnes responsables de cette publication.*
– *Bien, Nerret. Très bien. De mon côté, je contacte le proc'. Et tenez-moi au courant dès qu'il y a du neuf.*
– *Bien, commissaire. Bonne journée. »*

Voilà pour la mise en contexte. Cette petite histoire fictive pourra en rappeler d'autres, bien plus réelles. L'idée était simplement de montrer combien il est facile et rapide de *s'exposer* lors de la moindre connexion à Internet, et cela sans qu'aucune forme de surveillance ciblée ne soit nécessaire.

Quant à savoir quelles traces numériques permettent de remonter jusqu'à Alice et Benoît, l'un des objectifs de ce second tome est, justement, d'apporter des éclaircissements sur ces points. Avant de baliser, encore une fois, quelques pistes pour se protéger des attaques — ciblées ou non.

Chapitre 1

Bases sur les réseaux

Internet, ce n'est pas un espace virtuel, un nuage d'information abstrait où l'on trouve tout et n'importe quoi. En tout cas ce n'est pas seulement cela.

Ce qu'on appelle Internet est avant tout un ensemble de réseaux. Des millions de réseaux, agrégés sur plusieurs décennies et, de façon plus ou moins chaotique, gérés aussi bien par des entreprises, des universités, des gouvernements, des associations que des particuliers ; des millions d'ordinateurs et de matériaux de tous types, reliés entre eux par des technologies très diverses, allant du câble de cuivre à la fibre optique en passant par le sans-fil.

Mais pour nous, derrière notre petit écran, Internet c'est avant tout ce qu'il nous permet de faire : visiter des sites web, envoyer des emails, tchatter avec des gens ou télécharger des fichiers. De nouvelles applications apparaissent en permanence et seule l'imagination humaine semble en limiter les possibles.

Comprendre comment fonctionne Internet et comment s'y protéger c'est donc décortiquer un minimum cette complexité, afin de comprendre comment ces matériaux communiquent entre eux, mais aussi comment fonctionnent les diverses applications qu'on y utilise.

1.1 Des ordinateurs branchés entre eux

Assez tôt dans l'histoire de l'informatique il est apparu nécessaire, notamment dans le travail universitaire et dans le domaine militaire, de faire en sorte que des ordinateurs puissent partager des ressources ou des informations – et ce, à des distances de plus en plus grandes. Ainsi sont nés les réseaux informatiques. On a d'abord relié des ordinateurs les uns aux autres dans un lieu restreint – généralement une université, une entreprise ou un site militaire, puis on a relié ces lieux entre eux. Aux États-Unis, à la fin des années 60, est créé ARPANET (*Advanced Research Projects Agency Network*), un réseau qui reliait les universités dans tout le pays. Pour sa mise en place et son amélioration ont été inventées une bonne partie des techniques utilisées aujourd'hui avec Internet. La naissance d'Internet est liée à celle des logiciels libres, et il fonctionne selon des principes similaires d'ouverture et de transparence [1], ce qui n'empêche pas qu'au départ il a été développé pour répondre à des besoins militaires.

Les différents réseaux informatiques furent reliés au fur et à mesure, constituant ainsi progressivement Internet, qui se développe de façon importante depuis les années 90.

1. Selon Benjamin Bayard, « on ne peut pas dissocier Internet et logiciel libre » car ils sont apparus aux mêmes dates, avaient les mêmes acteurs, une croissance et un fonctionnement similaires. Benjamin Bayart, 2007, *Internet libre, ou Minitel 2.0 ?*, conférence aux 8e rencontres mondiales du logiciel libre à Amiens [https://www.fdn.fr/internet-libre-ou-minitel-2.html].

1.1.1 Un réseau d'ordinateurs

tome 1 ch. 1

« Un réseau est un ensemble de nœuds […] reliés entre eux par des liens » [2]. Dans un réseau informatique, les nœuds sont des ordinateurs. C'est donc un ensemble d'ordinateurs reliés entre eux par des câbles, des ondes, etc.

plus bas

Les ordinateurs qui font partie des réseaux ne ressemblent pas tous aux ordinateurs personnels, fixes ou portables que l'on utilise en général. Certains sont en effet spécialisés pour assurer des fonctions particulières au sein du réseau. Ainsi, la « box » qui permet à la plupart d'entre nous d'accéder à Internet est un petit ordinateur ; de même, les serveurs sur lesquels sont enregistrés les sites web sont aussi des ordinateurs. D'autres types d'ordinateurs spécialisés pourraient encore être ajoutés à cette liste : on en découvrira certains dans les pages qui viennent.

1.1.2 Carte réseau

tome 1 § 1.2

Malgré leurs différences, tous les ordinateurs connectés à un réseau ont nécessairement un point commun : en plus du matériel minimum qui compose un ordinateur, ils doivent disposer d'au moins un périphérique qui sert à se connecter au réseau. On l'appelle *carte réseau*. Elle permet d'établir le lien avec d'autres ordinateurs. De nos jours, plusieurs cartes réseau sont souvent intégrées dans tout ordinateur personnel (une filaire et une Wi-Fi par exemple).

tome 1 § 1.2.5

plus bas

Chaque carte réseau possède une adresse matérielle, qui l'identifie de façon plus ou moins unique. Dans la technologie filaire domestique, appelée Ethernet, comme dans la technologie sans-fil *Wi-Fi*, l'adresse matérielle est appelée *adresse MAC*. L'adresse MAC livrée avec la carte est conçue pour que la probabilité que deux cartes réseau possèdent la même adresse matérielle soit très faible [3], ce qui n'est pas sans poser problème en matière d'anonymat, comme nous le verrons plus loin.

page 28

1.1.3 Différents types de liens

Les façons les plus courantes de connecter des ordinateurs personnels en réseau sont soit d'y brancher un câble, que l'on appelle câble Ethernet, soit d'utiliser des ondes radio, avec le *Wi-Fi*.

Mais au-delà de notre prise téléphonique, nos communications sur Internet sont transportées par bien d'autres moyens. Il existe de nombreux supports pour transmettre l'information : câble de cuivre, fibre optique, ondes radio, *etc.* De la transmission par modem [4] des années 90 à la fibre optique [5] utilisée pour les connexions intercontinentales, en passant par l'ADSL [6] des années 2000, chacun d'eux a des caractéristiques différentes, notamment en termes de débit d'information (également appelé *bande passante*) et de coût d'installation et d'entretien.

Ces différentes technologies n'ont pas les mêmes faiblesses vis-à-vis de la confidentialité des communications qu'on leur confie ou des traces qu'elles laissent : il sera ainsi plus facile d'intercepter à distance un signal radio que de la lumière qui passe à l'intérieur d'une fibre optique.

2. Wikipédia, 2014, *Réseau informatique* [https://fr.wikipedia.org/wiki/Réseau_informatique]

3. Une adresse MAC se présente sous la forme d'une suite de 12 chiffres hexadécimaux (0 à 9 et a pour 10, b pour 11 et ainsi de suite jusqu'à f pour 15) comme par exemple `00:3a:1f:57:23:98`.

4. Modem est le mot condensé de *modulateur démodulateur* : il permet de transmettre des données numériques [tome 1 § 1.2.2] sur un canal permettant de véhiculer du son, comme par exemple une ligne téléphonique.

5. Une fibre optique est un fil constitué d'un matériau transparent permettant de transmettre des données sous forme d'impulsions lumineuses. Cela permet la transmission d'importants volumes d'information, même sur de longues distances.

6. L'ADSL (pour *Asymmetric Digital Subscriber Line*) est une technologie permettant de transmettre des données numériques [tome 1 § 1.2.2] sur une ligne téléphonique de manière indépendante du service téléphonique.

Un connecteur Ethernet standard RJ-45

1.2 Protocoles de communication

Pour que des machines puissent se parler, il ne suffit pas qu'elles soient reliées entre elles, il faut aussi qu'elles parlent une langue commune. On appelle cette langue un *protocole de communication*. La plupart des langues utilisées par les machines sur Internet sont définies de façon précise dans des documents publics : c'est ce qui permet à des ordinateurs et logiciels variés de fonctionner ensemble, pour peu qu'ils respectent ces standards. C'est ce que recouvre la notion d'*interopérabilité*.

1.2.1 Des fonctions complémentaires

Le fonctionnement d'Internet est basé sur l'utilisation de divers protocoles, répondant à différents besoins : le téléchargement d'un fichier, l'envoi d'un email, la consultation d'un site web, *etc.* fera intervenir différentes conventions, appelées *protocoles*.

Pour simplifier, nous détaillerons ci-dessous ces différents protocoles en les classant en trois catégories : protocoles physiques, réseau, puis applicatifs.

Et afin de bien comprendre, quoi de mieux qu'une analogie ?

Comparons donc le voyage de nos informations à travers Internet à l'acheminement d'une carte postale, dont les étapes, du centre de tri postal à la boîte aux lettres, correspondraient aux différents ordinateurs traversés.

Les protocoles physiques

Afin de livrer notre courrier à bon port, plusieurs moyens de transport peuvent être utilisés successivement : avion, bateau, camion, ou encore bicyclette.

Chacun de ces itinéraires obéit à un certains nombres de règlements : code de la route, aiguillage aérien, droit maritime, *etc.*

De même, sur Internet, les diverses technologies matérielles présentées précédemment impliquent l'usage de différentes conventions. On parle dans ce cas de *protocoles physiques*.

page précédente

Les protocoles réseau

Cependant, savoir naviguer ne suffit pas pour pouvoir acheminer notre carte postale. Il faut également savoir lire un code postal et posséder quelques notions de géographie pour atteindre le destinataire, ou du moins le centre de tri le plus proche.

C'est là qu'interviennent les *protocoles réseau* : leur but est de permettre l'acheminement d'informations d'une machine à une autre, parfois très éloignée, indépendamment des connexions physiques entre ces machines.

Les protocoles applicatifs

Maintenant que nous avons la possibilité de faire transiter des informations entre nos interlocuteurs, et de les faire parvenir à bon port, reste à s'accorder sur un dernier point : la langue utilisée sur la carte postale.

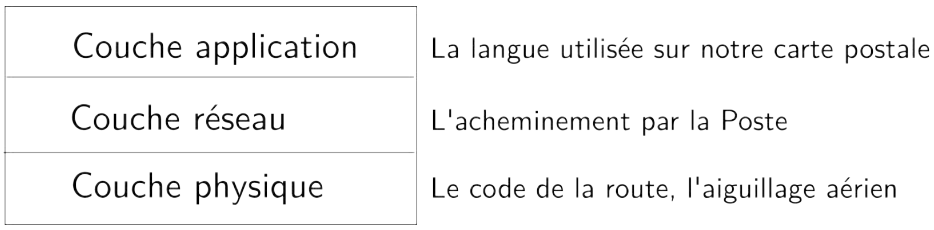
page 19

Aussi est-il nécessaire que, dans les échanges informatiques, les applications, utilisent une langue commune. Pour cela, elles s'accordent sur l'usage de *protocoles applicatifs* similaires.

Des protocoles encapsulés

En réalité, ces protocoles sont employés simultanément lors d'une communication, chacun d'entre eux ayant un rôle dans l'acheminement des informations.

Il est courant de représenter ces différents protocoles en couches qui se superposent.


Couche application
La langue utilisée sur notre carte postale
Couche réseau
L'acheminement par la Poste
Couche physique
Le code de la route, l'aiguillage aérien

Des protocoles encapsulés

De fait, lorsqu'on communique par courrier, notre communication se base sur l'écriture, puis sur l'acheminement par la Poste, qui se base elle-même sur différents moyens de transport.

De manière similaire, une application Internet utilisera un *protocole applicatif* précis, sera aiguillée grâce à l'usage de *protocoles réseau*, et parcourera les différentes infrastructures en respectant les *protocoles physiques* en vigueur.

On parle d'*encapsulation* (mettre dans une capsule) : le protocole applicatif est encapsulé dans le protocole de communication, protocole lui-même encapsulé dans le protocole réseau.

1.2.2 Le protocole IP

Il est intéressant de remarquer que contrairement aux protocoles physiques et applicatifs, les protocoles réseau sont relativement universels. Les protocoles physiques évoluent au gré des avancées technologiques, filaires ou sans-fil. Les protocoles applicatifs évoluent avec le développement de nouvelles applications : web, email, chat, *etc.* Entre ces deux niveaux, pour savoir par où passer et comment acheminer nos paquets à travers les millions de réseaux d'Internet, tout passe depuis les années 80 par le protocole *IP* : *Internet Protocol*.

Paquets

Dans le protocole *IP*, les informations à transmettre sont découpées et emballées dans des *paquets*, sur lesquels sont écrits notamment l'adresse d'expédition et celle de destination. Cette « étiquette » sur laquelle sont écrites les informations utiles à l'acheminement des paquets, à l'aller comme au retour, est appelée l'*en-tête* du paquet. Les paquets d'informations sont ensuite transmis indépendamment les uns des autres, parfois en utilisant différents chemins, puis réassemblés une fois arrivés à destination. C'est pourquoi un autre protocole, appelé *TCP*, pour *Transmission Control Protocol*, est couramment utilisé, en complément du protocole IP, pour s'assurer que tous les paquets sont arrivés et qu'ils sont rassemblés dans le bon ordre. Cela dit, il existe d'autres protocoles de transport de ces paquets [7].

Adresse IP

Pour que cela fonctionne, tout ordinateur connecté au réseau doit avoir une adresse, qui est utilisée pour lui envoyer des paquets : l'*adresse IP*. Cette adresse doit être unique au sein d'un réseau. En effet, si plusieurs ordinateurs du réseau avaient la même adresse, le réseau ne pourrait pas savoir à quel ordinateur envoyer les paquets.

On peut comparer l'adresse IP à un numéro de téléphone : chaque poste téléphonique doit avoir un numéro de téléphone pour qu'on puisse l'appeler. Si plusieurs postes téléphoniques avaient le même numéro de téléphone, il y aurait un problème.

Les adresses utilisées depuis les débuts d'Internet se présentent sous la forme de quatre numéros, séparés par un point : on parle d'adresses IPv4. Une adresse IPv4 ressemble à : `203.0.113.12`.

Ces adresses IPv4 ont été standardisées à une époque où les ordinateurs personnels n'étaient pas si répandus ; il y a 4 milliards d'adresses IPv4 possibles et cela semblait amplement suffisant, à tel point qu'elles ont été utilisées sans parcimonie. Depuis, le nombre d'adresses possibles est devenu trop petit par rapport à la quantité d'ordinateurs utilisés de nos jours. Pour cette raison, dans un futur proche, une nouvelle norme est supposée se généraliser, l'IPv6 [8], qui permet de connecter beaucoup plus d'ordinateurs à Internet [9] sans que deux d'entre eux ne se retrouvent avec la même adresse IP.

L'adresse IP est une information extrêmement utile pour quiconque cherche à surveiller ce qui se passe sur un réseau, car elle identifie un ordinateur du réseau de façon unique à un instant donné sans pour autant être une preuve réelle [10] contre une personne (un ordinateur peut être utilisé par plusieurs personnes). Elle peut néanmoins indiquer l'origine géographique d'une connexion, donner des indices, amorcer ou confirmer des suspicions.

1.3 Les réseaux locaux

On peut faire des réseaux sans Internet. D'ailleurs, les réseaux informatiques sont apparus bien avant Internet. Dans les années 60, des protocoles réseau comme HP-

7. Un autre protocole couramment utilisé pour la transmission est *UDP*, pour *User Datagram Protocol*, utilisé notamment quand il est nécessaire de transmettre des données très rapidement, quitte à en perdre une partie, comme pour la téléphonie sur Internet : les micro-coupures dans la communication entrainées par de légères pertes de données seront imperceptibles pour les utilisateurs, comme c'est le cas avec la téléphonie classique.

8. Une adresse IPv6 ressemble à : `2001:0db8:85a3:0000:0000:8a2e:0370:7334`.

9. Cette nouvelle norme pose de nouveaux problèmes vis-à-vis de notre anonymat en ligne. F. Florent, 2011, *Journal IPv6 et conséquences sur l'anonymat* [https://linuxfr.org/users/ffourcot/journaux/ipv6-et-conséquences-sur-lanonymat]. À suivre, donc…

10. legalis, 2013, *L'adresse IP, preuve insuffisante de l'auteur d'une suppression de données sur Wikipédia* [http://www.legalis.net/spip.php?page=breves-article&id_article=3885]

IB [11], ne permettant de connecter qu'un nombre restreint d'ordinateurs, faisaient déjà fonctionner des réseaux *locaux*.

1.3.1 Le réseau local, structure de base de l'Internet

Quand on branche plusieurs ordinateurs entre eux dans un même bâtiment, maison, école, université, bureau, *etc.*, on parle de *réseau local* (ou LAN, pour Local Area Network). Les ordinateurs peuvent alors communiquer entre eux, par exemple pour échanger des fichiers, partager une imprimante ou jouer en réseau.

On peut comparer les réseaux locaux aux réseaux téléphoniques internes de certaines organisations (entreprise, université).

Ces réseaux locaux sont souvent composés de différents appareils qui communiquent entre eux :

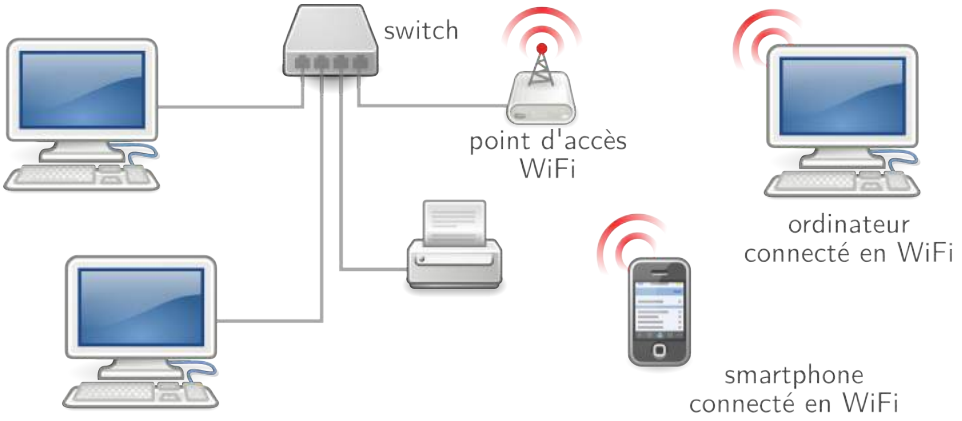


Schéma d'un réseau local

1.3.2 Le switch

Pour relier les machines constituant un réseau local, on les connecte en général chacune à une « multiprise » réseau, que ce soit avec un câble ou par des ondes Wi-Fi. On utilise souvent un « switch », que l'on peut comparer à une multiprise intelligente : au lieu de transmettre chaque paquet qui lui arrive à *tous* les ordinateurs branchés, un switch lit l'adresse indiquée sur le paquet pour ne l'envoyer qu'à la bonne prise de destination.

L'équivalent du switch des réseaux filaires s'appelle un « Point d'accès » dans le monde sans-fil. Chaque point d'accès possède un nom, qui est diffusé aux environs (c'est la liste des réseaux Wi-Fi qu'affiche notre logiciel réseau).

Pour reprendre notre comparaison, le switch est un peu comme la factrice de quartier, qui va dispatcher le courrier, dans tout le quartier, à chaque destinataire. Pour cela, le switch se souvient de la liste des cartes réseau, identifiées par leur adresse matérielle, branchées sur chacune de ses prises.

page 10

Tout comme l'accès physique à une machine donne beaucoup de possibilités pour récupérer les informations qui s'y trouvent, avoir accès physiquement à un réseau permet, sauf défenses particulières, de se faire passer pour l'une des autres machines de ce réseau. Cela rend possible de collecter beaucoup d'informations sur les communications qui y circulent, en mettant en place une attaque de type homme du milieu [12]. L'accès physique au réseau peut se faire en branchant un câble à un switch, mais aussi en y pénétrant *via* un point d'accès Wi-Fi.

page 64

11. Wikipédia, 2014, *HP-IB* [https://fr.wikipedia.org/wiki/HP-IB].
12. Vinc14 et junior0, 2013, *L'attaque de l'homme du milieu (MITM)* [http://fr.openclassrooms.com/informatique/cours/les-reseaux-de-zero/l-attaque-de-l-homme-du-milieu-mitm]

1.3.3 Adressage

Pour que les machines qu'on connecte au réseau puissent communiquer avec le protocole IP, elles doivent avoir chacune une adresse IP. Plutôt que de configurer l'adresse IP et les paramètres du réseau manuellement sur chaque machine, des logiciels et protocoles ont été développés pour automatiser cette étape lors du branchement à un réseau, comme par exemple le protocole DHCP [13].

page 12

Pour fonctionner, ce logiciel doit garder en mémoire l'association de telle carte réseau, identifiée par son adresse matérielle, à telle adresse IP. Ce logiciel fonctionne souvent sur un petit ordinateur qui peut se trouver dans le même boîtier que le switch (comme par exemple dans une « box » internet), mais il peut être ailleurs sur le réseau local. Cette correspondance entre l'adresse IP et l'adresse matérielle n'est utile que dans ce réseau local, car elle est liée au protocole physique utilisé en son sein. Ces adresses matérielles n'ont donc aucune raison technique de circuler sur Internet, mais cela arrive tout de même [14].

page 10

1.3.4 Connexion à d'autres réseaux

Si la plupart des réseaux locaux sont de nos jours reliés à Internet, cela n'est pas nécessaire : des ordinateurs peuvent tout à fait communiquer entre eux, au sein d'un réseau *local*, sans connexion à Internet. Ce qu'on appelle « Internet » n'est d'ailleurs qu'une interconnexion de réseaux locaux.

Pour connecter le réseau local à d'autres réseaux, il faut un *routeur*. C'est un ordinateur dont le rôle est de faire transiter des paquets entre deux réseaux ou plus. Quand ce *routeur* sert à faire transiter les paquets entre un réseau local et Internet, il est appelé *passerelle*.

Une « box » que l'on utilise pour raccorder une maison à Internet joue ce rôle de routeur. Elle dispose d'une carte réseau connectée au réseau local, mais aussi d'un modem ADSL connecté au réseau du fournisseur d'accès à Internet : on parle de modem-routeur. Elle fait partie non seulement du réseau local, mais aussi d'Internet : c'est l'adresse IP de la « box » qui est visible depuis Internet sur tous les paquets qu'elle achemine pour les ordinateurs du réseau local.

Le *fournisseur d'accès à Internet* (ou *FAI*) est une organisation offrant une connexion à Internet, que ce soit *via* une ligne téléphonique, un câble coaxial ou une fibre optique. En France, les principaux fournisseurs d'accès à Internet commerciaux sont, pour un usage domestique, Orange, Free, SFR ou numericable. Il existe aussi des FAI associatifs tels FDN.

La « box » est un petit ordinateur qui intègre, dans le même boitier que le modem-routeur, les logiciels permettant la gestion du réseau local (comme le logiciel de DHCP), ainsi qu'un switch Ethernet ou Wi-Fi pour brancher plusieurs ordinateurs mais aussi parfois un décodeur de télévision, un disque dur, *etc.*

plus bas

1.3.5 NAT et adresses réservées pour les réseaux locaux

Les organismes de standardisation d'Internet se sont rendus compte dans les années 90 que le nombre d'adresses IPv4 disponibles n'allait pas être suffisant pour faire face à la croissance rapide du réseau. Pour répondre à ce problème, on a réservé certaines plages d'adresses pour les réseaux privés, qui ne sont pas utilisées sur Internet : ce sont les *adresses privées*.

page 12

13. Utilisé dans les réseaux IPv4, DHCP signifie « protocole de configuration dynamique d'hôte » (*Dynamic Host Configuration Protocol* en anglais).
14. L'un des cas où l'adresse matérielle circule sur Internet est l'utilisation de portails captifs, dont on parlera plus tard [page 28].

Ainsi, la plupart des « box » assignent aux ordinateurs qui s'y connectent des adresses commençant par `192.168`[15]. Plusieurs réseaux locaux peuvent utiliser les mêmes adresses IP privées, au contraire des adresses IP sur Internet, qui doivent être uniques au niveau mondial.

Les paquets portant ces adresses ne peuvent pas sortir du réseau privé tels quels. Ces adresses privées ne sont donc utilisées que sur le réseau local : ma machine a l'IP `192.168.0.12` sur le réseau local, mais semblera utiliser l'adresse IP de la box pour les autres machines avec qui elle communiquera par Internet (par exemple, `203.0.113.48`) : ce sera l'*adresse publique*. La « box » se charge de modifier les paquets en conséquence grâce à la traduction d'adresse réseau (*NAT* pour *Network Address Translation*).

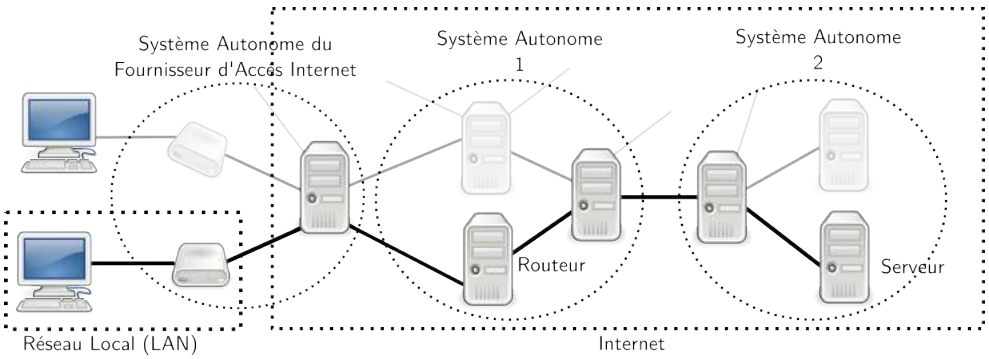
1.4 Internet : des réseaux interconnectés

Internet signifie *interconnected networks*, c'est-à-dire « interconnexion de réseaux ».

Comme son nom le suggère, le terme « Internet » désigne la connexion entre eux de nombreux réseaux de taille variable, comparables aux réseaux locaux dont on vient de parler. Chacun de ces réseaux est appelé *système autonome* (ou *AS*, pour *Autonomous System*).

1.4.1 Des systèmes autonomes

Un système autonome peut typiquement être celui d'un fournisseur d'accès à Internet (par exemple Free ou SFR). Chaque « box » qui sert à connecter un réseau local domestique à Internet fait ainsi partie du réseau du fournisseur d'accès, qui est lui-même interconnecté à d'autres systèmes autonomes pour former Internet. Les organisations qui hébergent des sites Internet (par exemple Dailymotion ou Google) et celles qui gèrent les « gros tuyaux », comme les câbles transatlantiques par lesquels passent une grande partie des flux de données de l'Internet, possèdent aussi leurs propres systèmes autonomes.



Internet est une inter-connexion de reseaux autonomes

Ainsi, Internet n'est pas un grand réseau homogène qui serait géré de façon centrale, mais est plutôt constitué d'un ensemble d'ordinateurs et de liaisons réseau qui appartiennent à des organisations et à des entreprises diverses et variées, chacune ayant son fonctionnement propre.

Tous ces réseaux, infrastructures et ordinateurs ne marchent pas tous seuls : ils sont gérés au quotidien par des gens, appelés *administrateurs ou administratrices systèmes*

15. Les plages d'adresses privées sont définies par convention dans un document appelé « RFC 1918 ». Elles incluent, en plus des adresses commençant par `192.168`, celles qui commencent par `10` et de `172.16` à `172.31`.

et réseau, ou « admins »[16]. Ce sont eux qui s'occupent d'installer, d'entretenir et de mettre à jour ces machines. Pour cela, ils ont *nécessairement* accès à énormément d'information sur les ordinateurs dont ils s'occupent.

En termes de surveillance, les intérêts commerciaux et les obligations légales des systèmes autonomes sont très variés en fonction des États et des types d'organisation en jeu (institutions, entreprises, associations, *etc.*). Personne ne contrôle entièrement Internet, et son caractère mondial rend compliquée toute tentative de législation unifiée. Il n'y a donc pas d'homogénéité des pratiques.

Interconnexion de réseaux

De la même façon que l'on a branché notre réseau local au système autonome de notre FAI, celui-ci établit des connexions à d'autres réseaux. Il est alors possible de faire passer des informations d'un système autonome à un autre. C'est grâce à ces interconnexions que nous pouvons communiquer avec les différents ordinateurs formant Internet, indépendamment du SA auquel ils appartiennent.

Un routeur

Un routeur est un ordinateur qui relie et fait communiquer plusieurs réseaux. Il est allumé en permanence et sa forme ressemble davantage à une grosse boîte de pizza qu'à un ordinateur personnel ; son principe de fonctionnement reste cependant similaire à celui des autres ordinateurs, et on lui adjoint quelques circuits spécialisés pour basculer très vite les paquets d'un réseau à un autre.

Les systèmes autonomes se mettent d'accord pour établir ces connexions au cas par cas, en général en fonction de leurs intérêts économiques : il s'agit pour eux d'envoyer leur trafic réseau à bon port au moindre coût. Cela passe souvent par des accord d'échange de trafic, qui découlent parfois sur des litiges : ainsi, en 2011, l'opérateur de transit Cogent et le fournisseur d'accès à Internet français Orange avaient un accord de « troc » de trafic réseau, mais Cogent envoyait jusqu'à 13 fois plus de trafic qu'Orange. Orange a alors ralenti volontairement le trafic réseau venant de Cogent, qui a à son tour porté plainte contre Orange… et perdu[17].

Des points d'interconnexion...

Les opérateurs fournissant l'infrastructure réseau ont d'abord commencé par tirer des câbles directement entre leurs routeurs, avant de se rendre compte que ça faisait beaucoup de câbles, et beaucoup de frais, et donc que ça serait souvent plus simple si tout le monde en tirait un qui arrivait au même endroit.

Il y a donc des endroits où de nombreux systèmes autonomes se relient entre eux. Chacun de ces endroits est appelé *point d'interconnexion* (*IX* pour Internet Exchange Point) : les systèmes autonomes qui veulent s'y connecter y amènent chacun un câble et y installent des routeurs. Du fait de la quantité importante de trafic qui passe par

16. On parlera plus loin d'« admins » pour désigner les administrateurs ou administratrices.
17. Marie-Cécile Renault, 2012, *Neutralité du Net : Orange gagne face à Cogent* [http://www.lefigaro.fr/hightech/2012/09/20/01007-20120920ARTFIG00732-neutralite-du-net-orange-gagne-face-a-cogent.php].

ces lieux, ceux-ci sont d'une grande importance stratégique pour les États et autres organisations qui voudraient surveiller ce qui transite par le réseau. [18]

...reliés entre eux

Les grands centres d'interconnexion sont reliés par de gros faisceaux de fibres optiques. L'ensemble de ces liaisons forment les *épines dorsales* (*backbones* en anglais) d'Internet.

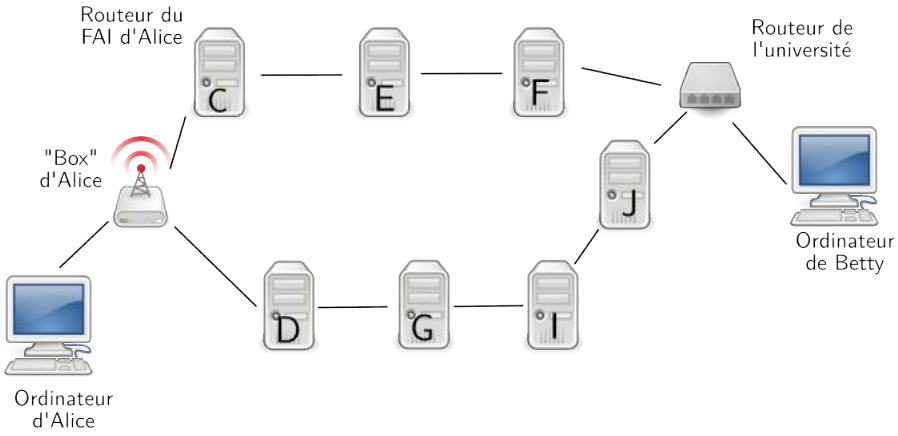
Ainsi, pour relier l'Europe à l'Amérique, plusieurs faisceaux de fibres optiques courent au fond de l'océan Atlantique. Ces faisceaux de fibres sont autant de points de faiblesse, et il arrive de temps en temps qu'un accident, par exemple une ancre de bateau qui coupe un câble, ralentisse fortement Internet à l'échelle d'un continent [19]. Ça peut paraître étrange, vu qu'historiquement, l'idée d'Internet était d'inspiration militaire : un réseau décentralisé, qui multiplie les liens pour être résistant à la coupure de l'un d'eux.

page 12

1.4.2 Routage

Nous avons vu que les ordinateurs s'échangeaient des informations en les mettant dans des paquets.

Imaginons deux ordinateurs connectés à Internet sur des réseaux différents qui veulent s'envoyer un paquet d'informations. Par exemple, l'ordinateur personnel d'Alice, situé en Europe, se connecte à celui de Betty, situé aux États-Unis.



Routage

L'ordinateur d'Alice accède à Internet par sa « box », qui se trouve sur le réseau de son fournisseur d'accès à Internet (ou FAI).

L'ordinateur de Betty, lui, fait partie du réseau de son université.

Le paquet destiné à l'ordinateur de Betty arrivera tout d'abord sur le réseau du FAI d'Alice. Il sera transmis au routeur C de son FAI, qui joue le rôle de centre de tri. Le routeur lit l'adresse de l'ordinateur de Betty sur le paquet, et doit décider à qui faire passer le paquet pour qu'il se rapproche de sa destination. Comment s'effectue ce choix ?

18. Guillaume Champeau, Juin 2013, *Comment l'Allemagne aussi espionne nos communications* [http://www.numerama.com/magazine/26279-comment-l-allemagne-aussi-espionne-nos-communications.html]

19. Pierre Col, 2009, *Internet, les ancres de bateaux et les séismes sous-marins* [http://www.zdnet.fr/blogs/infra-net/internet-les-ancres-de-bateaux-et-les-seismes-sous-marins-39602117.htm] (en français), Earl Zmijewski, 2008, *Mediterranean Cable Break* [https://www.renesys.com/blog/2008/01/mediterranean_cable_break.shtml] (en anglais).

Chaque routeur maintient une liste des réseaux auxquels il est connecté. Il envoie régulièrement les mises à jour de cette liste aux autres routeurs à qui il est branché, ses voisins, qui font de même. C'est grâce à ces listes qu'il peut aiguiller les paquets reçus et les transmettre vers leur destination.

Ainsi, le routeur du FAI d'Alice sait qu'il peut joindre le réseau de l'université de Betty par 4 intermédiaires en envoyant le paquet au routeur D. Mais il peut aussi l'envoyer par 2 intermédiaires, en le passant au routeur E. Il va choisir d'envoyer le paquet à E, qui a un chemin plus direct.

Le paquet arrive ainsi à E, le routeur d'un opérateur de transit, une organisation payée par le FAI d'Alice pour acheminer des paquets. E va faire le même genre de calcul, et envoyer le paquet à F. Le réseau de F comprend des ordinateurs non seulement en Europe, mais aussi aux États-Unis, reliés par un câble transatlantique. F appartient à une entreprise, similaire à celle qui gère E, qui est payée par l'université de Betty. F envoie finalement le paquet au routeur de l'université, qui l'envoie à l'ordinateur de Betty. Ouf, voilà notre paquet arrivé à destination.

Ainsi, chaque paquet d'information qui traverse Internet passe par plusieurs réseaux. À chaque fois, un routeur joue le rôle de centre de tri, et l'envoie à un routeur voisin. Au final, chaque paquet passe par beaucoup d'ordinateurs différents, qui appartiennent à des organisations nombreuses et variées.

De plus, la topologie du réseau, à savoir son architecture, la disposition des différents postes informatiques ainsi que leur hiérarchie changent au fil du temps. Lorsque le lendemain Alice se connecte de nouveau à l'ordinateur de Betty, les paquets que son ordinateur envoie ne prendront pas nécessairement le même chemin que la veille. Par exemple, si le routeur E est éteint à la suite d'une coupure de courant, le routeur du FAI d'Alice fera passer le paquet par D, qui avait auparavant une route plus longue.

C'est en agissant au niveau du routage que le gouvernement égyptien a fait couper Internet lors de la révolution de 2011. Les routeurs des principaux fournisseurs d'accès à Internet du pays ont cessé de dire aux autres routeurs que c'est à eux qu'il fallait s'adresser pour acheminer les paquets vers les ordinateurs égyptiens [20]. Ainsi, les paquets destinés à l'Égypte ne pouvaient plus trouver de chemin, interrompant de fait l'accès au réseau, le tout sans avoir coupé le moindre câble.

1.5 Des applications variées

On se sert souvent d'Internet pour accéder à des pages web, c'est-à-dire un ensemble de pages accessibles sur des serveurs, que l'on consulte à partir d'un navigateur web : `https://guide.boum.org` est un exemple de site web. Le langage courant confond fréquemment le web avec Internet, avec des expressions comme « aller sur Internet » par exemple. Or, le web n'est qu'un des nombreux usages d'Internet.

page 21

Il existe en fait de très nombreuses applications qui utilisent Internet, que la plupart des internautes n'ont pas conscience d'utiliser. Outre le web, on peut ainsi citer le courrier électronique, la messagerie instantanée, le transfert de fichiers, les monnaies numériques comme Bitcoin, *etc.*

Ainsi, vous pourrez rencontrer ces différents protocoles qui, s'ils utilisent Internet, ne sont *pas* du web :

- *SMTP*, *POP*, *IMAP* sont des protocoles utilisés dans la messagerie électronique [21], dont il existe également des versions chiffrées *IMAPS*, *POPS*, *SMTPS* ;

20. Stéphane Bortzmeyer, 2011, *Coupure de l'Internet en Égypte* [http://www.bortzmeyer.org/egypte-coupure.html].

21. Il existe une différence notable dans les protocoles employés, qui a des conséquences en termes de confidentialité et d'anonymat, selon qu'on utilise une boîte mail par le biais de son navigateur (webmail) ou par le biais d'un client de messagerie. Tout cela sera développé plus loin [page 93].

- *Skype*, *Yahoo Messenger*, *IRC* et *XMPP* sont des protocoles utilisés dans la messagerie instantanée.

En fait, une personne qui a des connaissances suffisantes en programmation peut créer elle-même un nouveau protocole et donc une nouvelle application d'Internet.

1.5.1 Protocole applicatif

page 11

Chaque application d'Internet utilise ainsi un langage particulier, appelé *protocole applicatif*, et met ensuite le résultat dans les paquets qui sont transmis par les protocoles réseau d'Internet. On peut comparer le protocole applicatif à la langue dans laquelle on écrit le texte d'une carte postale : il faut que l'expéditeur et le destinataire comprennent cette langue. Cependant, la Poste n'a pas besoin d'y comprendre quoi que ce soit, tant que la lettre contient une adresse valide.

tome 1 ch. 5

En général, les cartes postales ne sont pas mises dans des enveloppes : n'importe qui sur la route peut les lire. De même, les langages de la plupart des applications ne sont absolument pas chiffrés : non seulement la source et la destination écrites dans l'en-tête des paquets sont lisibles par quiconque, mais le contenu des paquets l'est aussi.

tome 1 § 4.1

Les protocoles applicatifs ne sont pas égaux non plus pour ce qui est de leur transparence. Si beaucoup d'entre eux sont définis par des conventions ouvertes, accessibles (et donc vérifiables) par tous, certaines applications utilisent des protocoles propriétaires pas ou peu documentés. Il est alors difficile d'analyser les éventuelles informations sensibles que contiendraient les données échangées. Par exemple, *Skype* fonctionne comme une véritable boîte noire, qui fait ce qu'on veut (communiquer), mais possiblement beaucoup d'autres choses : il a été notamment découvert que le contenu des messages est analysé et éventuellement censuré [22] et que toutes les adresses web qui sont envoyées *via* la messagerie sont transmises à Microsoft [23].

1.5.2 Port

On peut utiliser de nombreuses applications simultanément à partir d'un même ordinateur : lire ses emails dans le gestionnaire d'emails Thunderbird, regarder le site web de la SNCF, tout en tchattant avec ses potes par messagerie instantanée en écoutant de la musique en ligne. Chaque application doit recevoir seulement les paquets qui lui sont destinés et qui contiennent des messages dans une langue qu'elle comprend. Or, chaque ordinateur connecté au réseau n'a qu'une seule adresse IP. On ajoute donc, à cette adresse, un numéro, qui permet à l'ordinateur de faire parvenir le paquet à la bonne application. On écrit ce numéro sur le paquet, en plus de l'adresse : c'est le numéro de *port*.

Pour comprendre, comparons notre ordinateur à un immeuble : l'immeuble n'a qu'une seule adresse, mais abrite de nombreux appartements, et différentes personnes. Le numéro d'appartement inscrit sur une enveloppe permet de faire parvenir le courrier au bon destinataire. Il en est de même pour les numéros de port : ils permettent de faire parvenir les données à la bonne application.

Certains numéros de port sont assignés, par convention, à des applications particulières. Ainsi, quand notre navigateur veut se connecter à un serveur web, il sait qu'il doit toquer au port 80 (ou 443 dans le cas d'une connexion chiffrée). De la même façon, pour livrer un email, notre ordinateur se connectera en général au port 25 du serveur (ou 465 s'il s'agit d'une connexion chiffrée).

page ci-contre

22. Slate.com, 2013, *« Lance des œufs », « cinéma coquin »… La liste des mots surveillés par Skype en Chine* [http://www.slate.fr/monde/69269/tom-skype-surveillance-chine-espionnage-liste-noire].
23. Jürgen Schmidt, 2013, *Skype's ominous link checking : Facts and speculation* [http://www.h-online.com/security/features/Skype-s-ominous-link-checking-Facts-and-speculation-1865629.html] (en anglais).

Sur l'ordinateur qu'on utilise, chaque application connectée à Internet ouvre au moins un port, que ce soit un navigateur web, un logiciel de messagerie instantanée, un lecteur de musique, *etc.* Ainsi, le nombre de ports ouverts dans le cadre d'une connexion à Internet peut être très élevé, et fermer son navigateur web est souvent loin d'être suffisant pour couper toute connexion au réseau…

> Plus il y a de ports ouverts, plus il y a de points par lesquels s'infiltrer dans un ordinateur connecté au réseau. C'est le rôle habituellement dévolu aux **pare-feu** (**firewall** en anglais) que de ne laisser ouverts que certains ports définis dans leur configuration et de rejeter les requêtes allant vers les autres.

1.6 Des clients, des serveurs

Historiquement, dans les années 80, chaque ordinateur connecté à Internet fournissait une partie d'Internet. Non seulement il servait à « aller voir des choses sur Internet », mais il proposait également des informations, des données et des services aux autres utilisateurs connectés à Internet : il *faisait* Internet autant qu'il y *accédait*.

Le tableau général est très différent de nos jours. On a vu qu'il existe des ordinateurs allumés en permanence qui se chargent de relier des bouts d'Internet entre eux, les routeurs. De même il y a une autre catégorie d'ordinateurs allumés en permanence qui, eux, contiennent presque toutes les données et services disponibles sur Internet. On appelle ces ordinateurs des serveurs, car ils *servent* des informations et des services. Ils centralisent la plupart des contenus, que ce soient des sites web, de la musique, des emails, *etc.* Cela induit de la verticalité dans la hiérarchie du réseau. En effet, plus on dispose d'information, au sens large, plus on a potentiellement de pouvoir.

Les serveurs s'opposent aux clients, qui ne font qu'accéder aux informations. Cette situation correspond à un Internet où les utilisateurs deviennent des clients et sont donc principalement passifs, centralisant Internet autour des fournisseurs de contenus [24].

Prenons l'exemple d'un des services disponibles sur Internet, le site web du Guide d'autodéfense numérique [`https://guide.boum.org/`] : lorsque Alice consulte une page de ce site web, son ordinateur joue le rôle de *client*, qui se connecte au *serveur* qui héberge le Guide d'autodéfense numérique.

Cela dit, n'importe quel ordinateur peut être à la fois client et serveur, que ce soit dans un même temps ou successivement. Ces deux usages ne sont pas déterminés par le type de machine.

1.6.1 Les serveurs de noms

Lorsqu'Alice demande à son navigateur web d'aller sur le site du Guide d'autodéfense numérique, son ordinateur doit se connecter au serveur qui héberge ce site.

Pour cela, il est nécessaire de connaître l'adresse IP du serveur. Or une adresse IP est une suite de nombres assez pénible à mémoriser, taper ou transmettre : `204.13.164.188` (une adresse IPv4). Pour résoudre ce problème, il existe des serveurs à qui on peut poser des questions comme « quelle est l'adresse IP de `guide.boum.org` ? », comme on chercherait dans l'annuaire téléphonique quel est le numéro d'un correspondant. Ce système s'appelle le DNS (*Domain Name System*, ce qui donne « système de noms de domaine » en français). L'ordinateur d'Alice commence donc, *via*

page 12

24. La conférence de Benjamin Bayart, *Internet ou Minitel 2.0* (conférence aux 8e rencontres mondiales du logiciel libre, 13 juiller 2007, Amiens) [`http://www.fdn.fr/Internet-libre-ou-Minitel-2.html`] explique très bien ce glissement et les enjeux qu'il recouvre.

sa « box », par interroger le serveur DNS de son fournisseur d'accès à Internet pour obtenir l'adresse IP du serveur qui héberge le *nom de domaine* `guide.boum.org`.

L'ordinateur d'Alice reçoit en retour l'adresse IP du serveur et peut donc communiquer avec celui-ci.

1.6.2 Chemin d'une requête web

L'ordinateur d'Alice se connecte alors à cette IP, donc au serveur, et lui envoie une requête qui signifie « envoie-moi la page d'accueil du site web `guide.boum.org` ». Les paquets qui véhiculent la demande partent de son ordinateur et passent alors par sa « box » pour arriver au routeur de son fournisseur d'accès. Ils traversent ensuite plusieurs réseaux et routeurs, pour atteindre enfin le serveur de destination.

page 16
page 28

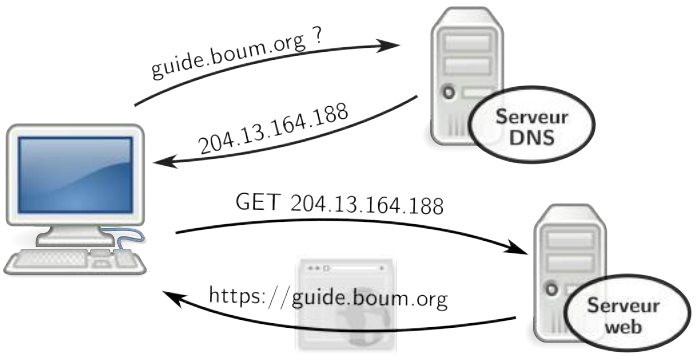


Schéma d'une requête web

1.6.3 Le logiciel serveur

Afin d'envoyer à Alice la page web demandée, le serveur recherche alors celle-ci dans sa mémoire, sur son disque dur, ou la fabrique :

Les pages consultables sur le web n'existent pas forcément sous une forme telle qu'on peut la voir sur notre ordinateur *avant* qu'on ait demandé à y accéder. Elle sont souvent générées automatiquement, à la demande. On parle alors de *site web dynamique* , par opposition au *site web statique* dont les pages sont écrites par avance.

tome 1 § 4.1.1

Par exemple, si l'on cherche « ouistiti moteur virtuose » dans un moteur de recherche, il n'a pas encore la réponse en réserve. Le serveur exécute alors le code source du site pour calculer la page contenant la réponse avant de nous l'envoyer.

Sur le serveur, il y a donc un logiciel qui fonctionne, et qui répond lorsqu'on lui fait une requête. Ce logiciel serveur est spécifique à chaque application : c'est lui qui comprend le protocole applicatif. Dans le présent exemple, ce logiciel recherche et sert à l'ordinateur d'Alice la page web : on l'appelle donc un serveur *web*.

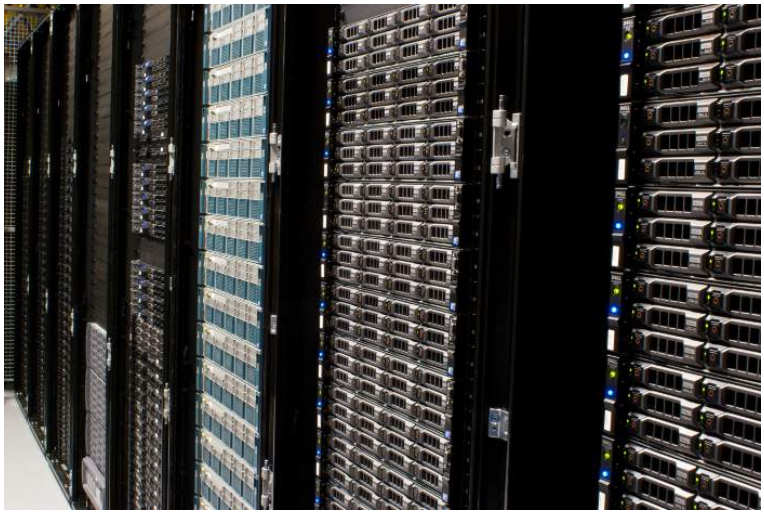
page 11

1.6.4 L'hébergement des serveurs

plus bas

Les serveurs, ordinateurs sur lesquels fonctionnent les logiciels serveurs évoqués précédemment, sont en général regroupés dans des immeubles disposant d'une bonne connexion au réseau et d'une alimentation électrique très fiable : des centres de données (ou *data center* en anglais).

De nos jours, la mode est de parler de *cloud computing* (« informatique en nuage » en anglais). Ce concept de marketing ne remet pas en cause la séparation entre clients et serveurs, bien au contraire. Il signifie simplement que les données sont susceptibles d'être déplacées d'un serveur à un autre, pour des raisons légales, techniques ou économiques. Et cela sans que leurs propriétaires en soient nécessairement informés.



Une allée de routeurs dans un centre de données

La société Google possède par exemple au moins 12 data centers répartis sur 3 continents [25] afin d'assurer l'opérabilité de ses services 24h/24h, 7j/7j, même lorsque certains équipements sont indisponibles.

Ce type d'hébergeur fait tourner des centaines de machines physiques réparties dans plusieurs centres de données autour du monde et mettent en commun leur puissance de stockage et de calcul pour en faire une super-machine abstraite. Ensuite, ils vendent des « machines virtuelles », c'est-à-dire des parts de puissance de calcul et de stockage de cette super-machine. L'« Amazon Elastic Compute Cloud » ou EC2 est l'un des services les plus connus dans ce domaine [26].

Une machine virtuelle peut être déplacée automatiquement en fonction de l'utilisation des machines physiques, de la qualité de leur connexion au réseau, *etc.* Avec une telle infrastructure, il est impossible de savoir à l'avance sur quelle machine physique et donc précisément à quel endroit se trouve une machine virtuelle donnée.

Cela rend en pratique impossible d'avoir du contrôle sur nos données [27]. Seront-elles réellement effacées des machines physiques si on les « supprime » ? On a vu dans le premier Tome qu' effacer des données sur un ordinateur était quelque chose de compliqué. Ce problème se corse encore si nous ne savons pas de quel ordinateur il s'agit. De plus, cela pose des problèmes juridiques : des données légales à un endroit peuvent se retrouver illégales parce que la machine qui les contient ou les sert sur Internet a changé de juridiction.

tome 1 § 4.3

Il y a donc eu un glissement d'un Internet où tout le monde consultait et distribuait des données, vers un modèle où les données étaient centralisées sur des machines physiques appelées serveurs, puis aujourd'hui vers le *cloud*, où ces mêmes données peuvent être enregistrées, parfois éparpillées, sur des serveurs indéterminés. Il devient extrêmement

25. Google, 2013, *Data center locations* [https://www.google.com/about/datacenters/inside/locations/index.html] (en anglais).
26. Wikipédia, 2014, *Amazon EC2* [https://fr.wikipedia.org/wiki/Amazon_EC2]
27. Jos Poortvliet, 2011, *openSUSE and ownCloud* [https://news.opensuse.org/2011/12/20/opensuse-and-owncloud/] (en anglais).

compliqué de savoir au final où elles sont réellement stockées, et l'utilisateur a encore moins de prise sur le devenir de ses données.

Chapitre 2

Traces sur toute la ligne

Le fonctionnement *normal* des réseaux implique que de nombreux ordinateurs voient ce que l'on y fait. Il n'est pas question ici de surveillance active. C'est parfois complètement nécessaire à leur fonctionnement. Il arrive aussi que ces informations soient collectées parce que c'est « plus pratique », par exemple pour diagnostiquer des problèmes.

Or, le fonctionnement de n'importe quel ordinateur laisse un certain nombre de traces. C'est le thème du premier tome de ce Guide. tome 1 ch. 2

Dans le cas d'une utilisation en ligne, ce n'est pas seulement l'ordinateur que l'on a devant les yeux qui peut garder des traces de ce que l'on fait sur le réseau, mais aussi chacun des ordinateurs par lesquels transitent les informations. Or ces informations circulent en général telles quelles, c'est-à-dire *en clair*, et non de façon chiffrée. tome 1 § 5.1

2.1 Sur l'ordinateur client

Le client utilisé pour se connecter au réseau a accès à tout ce que l'on y fait. Et comme lors de n'importe quelle autre utilisation, l'ordinateur en garde, bien souvent, des traces.

Comme cela a été longuement expliqué dans le premier tome de ce Guide, ces traces, et l'aisance avec laquelle elles peuvent être exploitées, dépendent très largement de l'ordinateur et du système d'exploitation utilisés. tome 1 ch. 2

2.1.1 La mémoire des navigateurs

Pour être plus agréables à utiliser, les navigateurs web enregistrent de nombreuses informations sur les pages que l'on consulte. Quelques exemples : la plupart des navigateurs web gardent un historique des pages web consultées ; ils proposent aussi souvent d'enregistrer ce que l'internaute saisit dans les formulaires qui se trouvent sur certaines pages web, ainsi que les mots de passe des différents comptes en ligne ; et ils enregistrent en général les pages récemment, ou courament, consultées, pour en accélérer le chargement : on parle de « mise en cache » [1]. C'est l'un des moyens pouvant permettre à la police de retracer notre navigation sur Internet. Souvenons-nous, dans notre histoire du début :

> – *Apparemment, sur l'ordinateur ayant servi à mettre en ligne les relevés bancaires, il y aurait eu une connexion à une boîte mail dont l'adresse correspond à une certaine Alice, chez Gmail, ainsi qu'une autre adresse*

1. Pour voir le contenu du cache du navigateur web *Firefox* ou de n'importe lequel de ses dérivés, comme *Iceweasel* ou le *Tor Browser Bundle*, taper `about:cache` dans la barre d'adresse.

email, chez no-log cette fois-ci, peu de temps avant la publication des documents incriminés.

2.1.2 Les cookies

Le mot Cookie vient de l'anglais « fortune cookie », en référence à des gâteaux qui cachent un message sur un petit papier. Un « cookie » est un petit « texte » envoyé par un site web que le navigateur de l'internaute stocke, puis renvoie au site à chaque visite. C'est ce qui permet par exemple aux applications de mail en ligne (« webmail ») de se rappeler qu'on est bien authentifié avec notre adresse et notre mot de passe pendant notre session, ou de mémoriser la langue que l'on désire utiliser.

Les cookies permettent aussi à un site web de pister les personnes qui le visitent.

Ainsi, les régies publicitaires sur Internet incluent, dans les publicités qu'elles affichent sur les sites, des cookies « traceurs » qui permettent de suivre l'internaute dans ses déplacements sur tous les sites qui affichent des publicités en provenance de la même régie publicitaire. Ainsi, elles peuvent « collecter des informations de plus en plus précises sur celui-ci et par conséquent lui proposer une publicité de mieux en mieux ciblée. » [2]

De plus, lorsqu'on consulte des pages web, celles-ci établissent en fait des connexions vers des sites de publicités et souvent vers les mêmes sites, ce qui augmente d'autant plus la possibilité de pistage de la part de ces sites.

Enfin, certains cookies ont une date d'expiration, mais d'autres sont à durée indéfinie – les sites qui nous les auront refilés pourront identifier notre navigateur pendant des années !

Les cookies classiques sont cependant restreints en termes de volume de données, et faciles à supprimer par un utilisateur averti. Aussi ont-ils été « améliorés » tout d'abord *via* la technologie *Flash* par un objet local partagé (en anglais *Local Shared Object* ou *LSO*), aussi appelé *cookie Flash*, qui permet de stocker plus de données [3].

La nouvelle norme *HTML5* inclut un mécanisme similaire, appelé « Stockage web local » [4].

D'autres techniques consistent à stocker le même cookie à différents emplacements dans le navigateur et à recréer à chaque visite ceux qui auraient été supprimés en partant du principe que si chacun peut être supprimé, ils ne le seront pas tous en même temps [5].

2.1.3 Applications côté clients

tome 1 § 4.1.1
page 22

Dans l'évolution du web et de ses navigateurs, il est rapidement devenu clair que pour avoir un minimum d'interactivité, il était nécessaire qu'une partie du code source du site web soit exécutée du côté du client, par le navigateur, et non sur le serveur web qui héberge le site.

Cela a plusieurs aspects pratiques : du côté du serveur web, c'est du travail en moins et des économies sur le matériel. Du côté du client, l'affichage et les fonctionnalités du site sont accélérés. Cela permet aussi de minimiser le trafic réseau entre le navigateur et le site web : plus besoin de demander une page complète du site à chaque fois que l'on clique sur un petit bouton, seul un petit fragment de la page doit être transmis.

2. CNIL, *La publicité ciblée en ligne*, communication présentée en séance plénière le 5 février 2009, M. Peyrat (rapporteur) [http://www.cnil.fr/PRIVOXY-FORCE/fileadmin/documents/La_CNIL/actualite/Publicite_Ciblee_rapport_VD.pdf]

3. Wikipédia, 2014, *Objet local partagé* [https://fr.wikipedia.org/wiki/Objet_local_partagé].

4. Simon K., 2012, *Stockage des données locales : Web Storage*, alsacréations [http://www.alsacreations.com/article/lire/1402-web-storage-localstorage-sessionstorage.html]

5. La bibliothèque JavaScript evercookie [http://samy.pl/evercookie/] (en anglais) est un exemple de ce type de technologies.

Des technologies ont été ajoutées aux navigateurs web pour permettre ces fonctionnalités : *JavaScript* et *Ajax*, *Flash* et *Java* en sont les principaux représentants.

Mais ces petits plus ont également un coût : comme précisé plus haut, cela signifie que l'auteur d'un site est en mesure d'exécuter le code de son choix sur l'ordinateur des personnes qui le visitent (ce qui pose de nombreux problèmes de sécurité, comme nous l'avons vu dans le premier tome de ce guide). Bien sûr, des protections ont été mises en place au sein des navigateurs [6], mais elles ne couvrent pas tous les risques et ne remplacent en tout cas pas la vigilance des internautes [7].

tome 1 § 3.1

D'autant que ces technologies ont parfois des fonctionnalités qui, si elles peuvent être utiles, posent question : ainsi, Flash ou WebRTC [8] peuvent accéder au micro et à la caméra de l'ordinateur sur lequel ils sont exécutés [9]. Et dans le cas de Flash, il s'agit d'un logiciel propriétaire… L'usage de Flash est également problématique car l'intérieur même du moteur d'exécution ne peut être inspecté, et les corrections de trous de sécurité ne peuvent être faites *que* par la société Adobe qui le distribue.

tome 1 § 4.1.2

On a vu que placer sa confiance dans un logiciel était un choix complexe. Dès lors, l'exécution de ce genre de programmes pose des questions quant au pouvoir donné aux auteurs de sites ou d'applications web d'accéder aux ressources de notre ordinateur, et aux informations qu'il contient.

tome 1 § 4.1

De plus, avant d'être exécutés par le navigateur, ces bouts de code transitent par le réseau, souvent sans aucune authentification. Cela laisse le loisir aux personnes malintentionnées et bien placées de les modifier, tout comme le reste d'une page web. Pour y introduire, par exemple, un logiciel malveillant. Il est aussi possible de jouer avec les données que ces codes doivent traiter pour tenter de détourner leur usage. Ce genre de manipulation de pages web a par exemple été détecté par le passé lors de l'utilisation du point d'accès Wi-Fi d'un hôtel à New York qui utilisait un équipement réseau dédié à cette tâche. [10]

tome 1 ch. 3

Au final, un navigateur web moderne a tellement de fonctionnalités qu'un éventuel adversaire dispose d'un nombre considérable d'angles d'attaque.

2.1.4 Dans les journaux des logiciels

Le navigateur web n'est pas le seul logiciel à enregistrer des traces sur l'ordinateur utilisé ; la plupart des logiciels ont des journaux.

tome 1 § 2.4

Par exemple, les logiciels de messagerie instantanée enregistrent souvent l'historique des conversations ; les logiciels de P2P ou Torrent, eux aussi, ont tendance à se souvenir de ce qu'on a téléchargé récemment ; les logiciels de mail gardent les emails qu'on a téléchargés, *etc.*

> – *Apparement, les collègues ont fini par retrouver le document sur un des ordinateurs. Il a été téléchargé depuis le navigateur, et modifié.*

Dans notre histoire, ce sont les journaux de logiciels tels que le navigateur web et l'éditeur de texte qui ont permis de retrouver les trace du document de Benoît.

6. Il s'agit en général de ne donner accès au code des sites web qu'à des fonctions limitées en l'exécutant dans un « bac à sable ». (Wikipédia, 2014, *Sandbox (sécurité informatique)* [https://fr.wikipedia.org/wiki/Sandbox_(sécurité_informatique)]

7. Félix Aimé, 2012, *Sécurité des navigateurs* [http://free.korben.info/index.php/Sécurité_des_navigateurs]

8. Technologie qui vise à intégrer aux navigateurs web les communications en temps réel, par exemple la voix sur IP (VOIP).

9. Une faille de sécurité dans Flash permettait à un pirate de déclencher à leur insu la webcam des personnes qui visitent un site web. Vincent Hermann, 2011, *Flash corrigé pour empêcher l'espionnage par webcam et micro*, PC INpact [http://www.pcinpact.com/news/66557-adobe-flash-correction-faille-webcam-espion.htm]

10. Justin Watt, 2012, *Hotel Wifi JavaScript Injection* [http://justinsomnia.org/2012/04/hotel-wifi-javascript-injection/] (en anglais).

2.2 Sur la « box » : l'adresse matérielle de la carte réseau

page 10

On a vu que la carte réseau utilisée par tout ordinateur pour se connecter possède une adresse matérielle, ou adresse MAC. Cette adresse est utilisée par les équipements réseaux pour rediriger un paquet de données vers la bonne carte réseau, lorsque plusieurs ordinateurs sont connectés sur la même « box » par exemple.

Normalement, cette adresse ne sort pas du réseau local. Cependant, on se connecte en général directement à la « box » d'un fournisseur d'accès à Internet. Chaque carte réseau connectée à la « box » lui donne donc son adresse matérielle.

Le plupart des « box » gardent un journal qui contient ces adresses matérielles, au moins pendant le temps où elles sont allumées. Elles ne sont pas supposées laisser fuiter ce journal. Cependant, il est difficile de savoir les types et la quantité d'informations contenues dans ce journal, ainsi que l'existence potentielle de portes dérobées [11] ou de failles de sécurité permettant d'y accéder. En effet, ces « box » fonctionnent avec un logiciel installé par le fournisseur d'accès à Internet, qui y garde un accès privilégié, ne serait-ce que pour effectuer les mises à jour du logiciel. Pour nous, la « box » est donc à considérer comme une véritable boîte noire, dont nous n'avons pas les clés, qui peut connaître (et faire) beaucoup de choses sur le réseau local.

tome 1 § 1.4

De plus, lorsque le réseau local inclut l'usage du Wi-Fi, il se peut que de manière plus ou moins accidentelle les adresses matérielles des ordinateurs se connectant à la « box » en Wi-Fi soient enregistrées par d'autres ordinateurs écoutant ce qui « passe dans les airs ». C'est ainsi que les Google Cars, en même temps qu'elles parcouraient des milliers de rues pour établir la carte de Google Street View, en ont profité pour « capturer » les adresses MAC des ordinateurs environnants [12].

Il est par contre possible de changer temporairement l'adresse matérielle d'une carte réseau, afin par exemple de ne pas être pistés avec nos ordinateurs portables [13] lors de nos déplacements.

Il faut aussi mentionner les cas où, avant de pouvoir se connecter à Internet, on doit entrer un *login* et un mot de passe dans son navigateur web : c'est souvent le cas sur les réseaux Wi-Fi publics, que ce soit ceux d'une agglomération, d'une institution ou d'un fournisseur d'accès à Internet (*FreeWifi*, *SFR WiFi public* et autres *Bouygues Telecom Wi-Fi*). On appelle ces pages des *portails captifs*. Dans ce cas, en plus de l'adresse matérielle de la carte Wi-Fi, on donne à l'organisation qui gère le portail l'identité de la personne abonnée correspondant à ces identifiants.

2.3 Sur les routeurs : les en-têtes de paquets

page 16

Sur le chemin entre un ordinateur et le serveur auquel on souhaite se connecter, il y a de nombreux routeurs, qui relaient les paquets et les envoient au bon endroit.

Pour savoir où envoyer un paquet, ces routeurs lisent une sorte d'enveloppe sur laquelle un certain nombre d'informations sont écrites ; on appelle cette « enveloppe » l'*en-tête* du paquet.

page 12

L'en-tête d'un paquet contient de nombreuses informations qui sont nécessaires à son acheminement, et notamment l'adresse IP du destinataire, mais aussi celle de l'expéditeur (à qui la réponse devra être envoyée). Le routeur voit donc quel ordinateur veut parler à quel autre ordinateur, de la même manière que le facteur doit avoir l'adresse du destinataire pour lui transmettre le courrier, ainsi que l'adresse de l'expéditeur pour un éventuel retour.

11. Une revue détaillée de nombreux routeurs compromis est disponible dans l'article Tiger-222, 2013, *Routeurs déchus* [http://tiger-222.fr/?d=2013/10/29/23/59/51-routeurs-dechus].
12. TOMHTML, 2011, *Google condamné, l'analyse des Google cars dévoilée* [http://www.zorgloob.com/2011/03/21/google-street-view-cnil/].
13. Wikipédia, 2014, *Mac Spoofing* [https://fr.wikipedia.org/wiki/Filtrage_par_adresse_MAC#MAC_Spoofing].

Les en-têtes contiennent aussi le numéro du port source et celui du port de destination, ce qui peut renseigner sur l'application utilisée. page 20

Pour faire leur travail, les routeurs *doivent* lire ces informations ; ils *peuvent* aussi en garder la trace dans des journaux.

Bien qu'ils n'aient pas de bonne raison de le faire, les routeurs sont aussi en mesure d'accéder à l'*intérieur* de l'enveloppe transportée ; par exemple, le contenu de la page web consultée par un internaute, ou celui d'un email envoyé : on parle alors d'examen approfondi des paquets [14] (*Deep Packet Inspection* ou DPI en anglais). page 45

Le fournisseur d'accès à Internet français Orange inclut par exemple dans le contrat de ses abonnés une clause concernant l'usage des « données relatives » à son trafic [15].

2.4 Sur le serveur

Le serveur a accès comme les routeurs aux en-têtes des paquets IP et donc à toutes ces informations dont on vient de parler. Il regarde notamment l'adresse IP de la box utilisée par l'ordinateur qui se connecte pour savoir à qui envoyer la réponse. page 12

En plus des en-têtes IP, correspondant à la couche réseau de la communication, le serveur lira les en-têtes de protocole applicatif qui correspond à la couche applicative de la communication. page 11 page 20

Mais le serveur lit aussi le contenu des paquets eux-mêmes : c'est en effet lui qui doit ouvrir l'enveloppe et lire la lettre pour y répondre. Le logiciel serveur va alors interpréter la lettre reçue, qui est écrite avec le protocole applicatif, pour fournir la réponse adaptée.

Or, de très nombreux protocoles applicatifs véhiculent aussi des informations qui permettent d'identifier l'ordinateur qui se connecte - c'est ce que nous allons voir en détails ici.

Les serveurs ont, comme les ordinateurs clients, des journaux systèmes — on en parlera davantage dans la partie suivante. page 33

2.4.1 Les en-têtes HTTP

Lorsqu'un navigateur demande une page web, il inclut dans la requête le nom du logiciel, son numéro de version, le système d'exploitation utilisé et la langue dans laquelle celui-ci est configuré.

Voici une requête envoyé par le navigateur web *Firefox* :

```
GET /index.html HTTP/1.1
Host :www.exemple.org
User-Agent :Mozilla/5.0 (X11; Linux x86_64; rv :24.0) Gecko/20140429 Firefox/24.0
↪    Iceweasel/24.5.0
Accept :text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language :fr,fr-fr;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding :gzip,deflate
Connection :keep-alive
Referer :https ://www.google.com/search?q=example+domain&ie=utf-8&oe=utf-8&aq=t
Cookie :PHPSESSID=r2t5uvjq435r4q7ib3vtdjq120
```

14. Wikipédia, 2014, *Deep Packet Inspection* [https://fr.wikipedia.org/wiki/Deep_packet_inspection]
15. Martin Untersinger, 2012, *Fin de l'Internet illimité : ça se précise chez Orange, qui dément* [http://www.rue89.com/2012/10/11/fin-de-linternet-illimite-ca-se-precise-chez-orange-236102].

On y voit tout d'abord une commande contenant la nom de la page demandée (`index.html`), le nom de domaine correspondant (`www.exemple.org`), suivie d'un en-tête qui contient entre autres le nom et la version de navigateur (`Mozilla/5.0 Gecko/20140429 Firefox/24.0 Iceweasel/24.5.0`) ainsi que le système d'exploitation utilisé (`Linux x86_64`), les langues acceptées (`fr-fr` pour français de France, `en-us` pour anglais des États-Unis), la page sur laquelle se trouvait le lien que l'internaute a suivi pour arriver à la page demandée (`https://www.google.com/search?q=example+domain&ie=utf-8&oe=utf-8&aq=t`, noter les termes de recherche : « exemple » et « domaine »), et le cookie de session (`PHPSESSID=r2t5uvjq435r4q7ib3vtdjq120`).

page 26

Ces informations sont là pour être utilisées par le serveur web, qui va adapter sa réponse en fonction : c'est notamment grâce à cela qu'un site disponible en plusieurs langues s'affiche dans notre langue sans que nous ayons eu à l'indiquer.

Mais ces informations, comme toutes celles qui transitent par le serveur, sont aussi accessibles aux personnes qui s'occupent de la maintenance du serveur : ses admins… et leur hiérarchie. En général, les serveurs gardent aussi ces informations dans des journaux, plus ou moins longtemps, notamment pour faire des statistiques et pour faciliter les diagnostics en cas de panne. Ils ajoutent aux en-têtes l'adresse IP d'origine ainsi que la date et l'heure. Voici une ligne de journal enregistrée pour notre requête (l'adresse IP se trouve au début : `203.0.113.16`) :

page 36

```
203.0.113.16 - - [01/Jan/2010 :00 :00 :00 +0100] "GET /page.html HTTP/1.1" 200 9042
↪    "http ://www.exemple.org/index.html" "Mozilla/5.0 (Windows ; U ; Windows NT
↪    6.1 ; en-US ; rv :1.9.2.3) Gecko/20100401 Firefox/3.6.3"
```

2.4.2 Les en-têtes email

Chaque courrier électronique inclut un en-tête ; malgré son nom, ce dernier n'a strictement rien à voir avec l'en-tête d'une page web. Cet en-tête contient des informations sur les données contenues dans l'email : un autre exemple de méta-données, les « données sur les données ». Il est rarement montré dans sa totalité par notre logiciel de courrier électronique, mais il reste néanmoins bien présent. Il inclut souvent de nombreuses informations sur l'expéditeur - bien plus que son adresse email.

tome 1 § 2.6

Dans l'exemple suivant, on peut lire l'adresse IP publique, à savoir celle qui sera visible sur Internet, de l'ordinateur utilisé pour envoyer l'email (`203.0.113.98`), ce qui permet de connaître l'endroit où l'expéditeur se trouvait à ce moment-là, l'adresse IP de son ordinateur à l'intérieur de son réseau local (`192.168.0.10`), le logiciel de mail utilisé (Icedove 24.4.0) ainsi que le système d'exploitation (Linux) et le type de machine (i686) :

page 15

```
Return-Path :<betty@fai.net>
Delivered-To :alice@exemple.org
Received :from smtp.fai.net (smtp.fai.net [198.51.100.67])
        by mail.exemple.org (Postfix) with ESMTP id 0123456789
        for <alice@exemple.org> ; Sat, 1 Jan 2014 20 :00 :00 +0100 (CET)
Received :from [192.168.0.10] (paris.abo.fai.net [203.0.113.98])
        by smtp.fai.com (Postfix) with ESMTP id ABCDEF1234 ;
        Sat, 1 Jan 2014 19 :59 :49 +0100 (CET)
Message-ID :<CB0ABB91.17B7F@fai.net>
Date :Sat, 1 Jan 2014 19 :59 :45 +0100
From :Betty <betty@fai.net>
User-Agent :Mozilla/5.0 (X11 ; U ; Linux i686 ; en-US ;
    rv :1.9.1.16) Gecko/20111110 Icedove/24.4.0
MIME-Version :1.0
```

```
To :Alice <alice@exemple.org>
Subject :À mardi
Content-Type :text/plain; charset=iso-8859-1
Content-Length :22536
Lines :543
```

Ces en-têtes contiennent aussi parfois l'identifiant de l'abonné chez son prestataire d'email ou le nom de sa machine [16].

À l'instar de ces quelques exemples courants, quasiment toutes les applications envoient des informations sur le contenu, mais aussi des méta-données dans leur protocole.

tome 1 § 2.6

2.5 Les traces qu'on laisse soi-même

Il n'y a pas que les traces que laisse le fonctionnement des réseaux : il y a bien sûr aussi celles que nous laissons nous-mêmes, de façon plus ou moins volontaire, par exemple en saisissant des informations sur des sites web ou simplement en nous connectant à des services.

Tenter de maîtriser les traces qu'on laisse sur les réseaux, c'est donc aussi réfléchir aux utilisations qu'on fait des services proposés sur Internet, et aux données qu'on leur confie – des thèmes qu'on traitera plus avant dans les parties à venir.

16. La plupart du temps, cela se trouve dans la ligne `Received` de la première machine ou dans le `Message-Id`. Mais certains autres logiciels ou services de messagerie rajoutent d'autres lignes plus spécifiques.

Chapitre 3

Surveillance et contrôle des communications

Au-delà des traces laissées par le fonctionnement même des réseaux en général et d'Internet en particulier, il est possible d'« écouter » nos activités sur Internet à plusieurs niveaux. De plus en plus souvent, les organismes qui font fonctionner des parties d'Internet (câbles, serveurs, *etc.*) sont même dans l'obligation légale de conserver un certain nombre de données sur ce qui se passe sur leurs machines, au titre de lois de *rétention de données*.

3.1 Qui veut récupérer les données ?

Diverses personnes ou organisations peuvent porter des regards indiscrets sur les échanges *via* Internet. Parents un peu trop curieux, sites web à la recherche de consommateurs à cibler, multinationales comme Microsoft, gendarmes de Saint-Tropez, ou *National Security Agency* états-unienne… Comme dans le cas des mouchards sur les ordinateurs personnels, les différentes entités impliquées ne collaborent pas forcément ensemble, et ne forment pas une totalité cohérente. Si les curieux sont trop variés pour prétendre dresser une liste exhaustive des intérêts en jeu, on peut toutefois décrire quelques motivations parmi les plus courantes.

tome 1 ch. 3

3.1.1 Des entreprises à la recherche de profils à revendre

« Vous décidez de réserver un billet d'avion pour New-York sur Internet. Deux jours plus tard, en lisant votre quotidien en ligne, une publicité vous propose une offre intéressante pour une location de voitures à New York. Ce n'est pas une simple coïncidence : il s'agit d'un mécanisme de publicité ciblée, comme il s'en développe actuellement de plus en plus sur Internet. » [1].

La publicité est l'une des principales sources de revenus des entreprises qui fournissent des services « gratuits » sur Internet : boîtes mails, moteurs de recherche, médias sociaux, *etc.* Or, du point de vue des annonceurs, la qualité et donc le prix d'un espace publicitaire en ligne est fonction de l'intérêt que les internautes vont porter aux publicités.

Dès lors, les données personnelles valent de l'or. Centres d'intérêt, sexe, âge, *etc.* : autant d'informations qui permettent de présenter les publicités auxquelles l'internaute est le plus susceptible de réagir. Ainsi, Gmail, le service d'email de Google, utilise-t-il le résultat de l'analyse du contenu des emails pour afficher des publicités

1. CNIL, 2009, *La publicité ciblée en ligne* [http://www.cnil.fr/fileadmin/documents/La_CNIL/actualite/Publicite_Ciblee_rapport_VD.pdf].

correspondantes [2] : pour prendre un exemple (authentique !), une personne en train de se séparer de son partenaire pourra voir s'afficher, en marge de ses emails, des publicités pour des sites de rencontres…

En outre, chaque site visité est un « centre d'interêt » de plus. En additionnant ces informations les unes aux autres, c'est tout un profil qui se dessine. Un petit logiciel permet de voir quels cookies se téléchargent sur notre ordinateur à chaque page consultée [3]. Si l'internaute commence par visiter allocine.fr, quatre régies publicitaires enregistrent sa visite. Se rendant ensuite sur le site du Monde ce sera quatre régies qui seront au courant, dont deux qui se trouvaient déjà sur le site d'*allociné*. Elles savent donc que l'internaute a visité ces deux sites et peut donc recouper ces deux centres d'intérêt. En se rendant par la suite sur deux autres sites (*Gmail* et dailymotion), ce sont au total 21 régies publicitaires qui ont eu connaissance de la visite de cet utilisateur. Dans chacune de ces visites se trouvaient les régies publicitaires XiTi et Google-Analytics. Le plus gros moteur de recherche a donc eu connaissance de la totalité des sites visités et peut maintenant mettre en place sa publicité ciblée.

page 26

Les médias sociaux sont particulièrement bien placés pour obtenir directement des internautes des données personnelles les concernant. Ainsi, sur Facebook, un annonceur peut « cibler une publicité auprès des jeunes de 13 à 15 ans habitant Birmingham en Angleterre et ayant "la boisson" comme centre d'intérêt. De plus, Facebook indique que la cible choisie comporte approximativement une centaine de personnes [4]. La société Facebook exploite ainsi les données qu'elle collecte de ses membres de manière à fournir une publicité qui peut être très ciblée » [5].

La publicité ciblée est d'ailleurs « l'une des raisons qui a poussé les acteurs Internet à diversifier leurs services et leurs activités, afin de collecter toujours plus d'informations sur le comportement des utilisateurs sur Internet. » « Par exemple, Google fournit des services de recherche. Il a racheté des sociétés de publicité comme DoubleClick. Il a récemment lancé un service Google Suggest, intégré à son navigateur Chrome, qui envoie à Google l'ensemble des pages web visitées par les internautes, même quand ces derniers n'y ont pas accédé *via* le moteur de recherche, *etc.* » [6]

Pour se donner une idée de l'importance des enjeux, notons que Google a racheté la société Doubleclick pour la somme de 3,1 milliards de dollards [7].

Cette accumulation de données et leur traitement permet également à Google de trier et d'adapter les résultats aux supposés centres d'intérêt de l'internaute. Ainsi, pour une recherche identique, deux personnes ayant un profil différent n'obtiendront pas le même résultat, ce qui a pour effet de renforcer chaque personne dans ses propres intérêts et ses propres convictions. C'est ce que certaines personnes nomment « l'individualisation de l'Internet » [8].

2. « Dans Gmail, le ciblage des annonces est entièrement automatisé : personne ne consulte vos emails ni les informations relatives à votre compte Google avant de vous proposer des annonces ou des informations connexes. Si vous ne désactivez pas cette option, les annonces contextuelles en rapport avec le message que vous êtes en train de lire, [..] risquent de s'afficher. » Google, 2014, *Annonces dans Gmail et vos données personnelles* [https://support.google.com/mail/answer/6603?hl=fr&ref_topic=3394525].

3. Clochix, 2011, *Collusion, pour visualiser comment nous sommes tracés en ligne* [http://www.clochix.net/post/2011/07/10/Collusion,-pour-visualiser-comment-nous-sommes-tracés-en-ligne].

4. Une interface similaire est disponible publiquement et permet de répondre à des requêtes inquiétantes : Tom Scott, 2014, *Actual Facebook Graph Searches* [http://actualfacebookgraphsearches.tumblr.com/] (en anglais).

5. CNIL, *La publicité ciblée en ligne* (op. cit.), p. 13.

6. CNIL, *La publicité ciblée en ligne* (op. cit.), p. 4.

7. Le Monde, 2007, *Google rachète DoubleClick pour 3,1 milliards de dollars* [http://www.lemonde.fr/technologies/article/2007/04/14/google-rachete-doubleclick-pour-3-1-milliards-de-dollars_896316_651865.html].

8. Xavier de la Porte, 2011, *Le risque de l'individualisation de l'Internet*, InternetActu.net, Fondation Internet nouvelle génération [http://www.internetactu.net/2011/06/13/le-risque-de-lindividualisation-de-linternet].

En plus d'être ciblée thématiquement, la publicité l'est aussi géographiquement : grâce aux GPS intégrés dans les terminaux mobiles tels les smartphones, mais aussi grâce à l'adresse IP et aux réseaux Wi-Fi « visibles » à portée de l'ordinateur portable ou du téléphone [9]. Ainsi, il est par exemple possible de faire apparaître des publicités pour des boutiques situées à proximité de l'abonné.

Des intérêts économiques poussent ainsi les fournisseurs de services Internet à rassembler des profils d'internautes, les plus précis possibles, pour ensuite vendre, directement ou pas, des espaces publicitaires ciblés.

Une fois ces informations rassemblées, les entreprises en question ne rechigneront en général pas à les communiquer aux flics s'ils les leur demandent. Tous les gros fournisseurs de contenus ont des bureaux dédiés à répondre aux requêtes et donc des formulaires, procédures, *etc.* écrites pour les flics, pour expliquer la meilleure marche à suivre pour demander des informations [10].

3.1.2 Des entreprises et des États cherchant à préserver leurs intérêts

D'autres entreprises s'intéressent à ce qui se passe sur Internet pour préserver leurs intérêts. Cela va de la lutte menée par l'industrie de l'audiovisuel contre le téléchargement illégal à la veille technologique : les entreprises observent et analysent en temps réel et de manière automatisée des centaines de sources (sites d'actualité, bases de dépôt de brevets, blogs d'experts…) afin de connaître rapidement les dernières avancées technologiques et de rester les plus compétitives possible.

Les entreprises sont loin d'être les seules à scruter Internet. Les États, de la justice aux services secrets en passant par les différents services de police sont même certainement les plus curieux.

De plus en plus de pays mettent en place des lois visant à rendre possible l'identification des auteurs de toute information qui circule sur Internet [11].

Mais cela va plus loin encore. Les agences de renseignement et autres services secrets ne se contentent plus d'espionner quelques groupes ou personnes qu'elles considèrent comme des cibles. À la limite de la légalité, la NSA, agence de renseignement étatsunienne, collecte « toutes sortes de données sur les personnes – nous pensons que cela concernerait des millions de personnes » [12]. Parmi ses objectifs : « examiner " quasiment tout ce que fait un individu sur Internet " » [13] et établir un *graphe social*, c'est-à-dire « le réseau de connexions et de relations entre les gens » [14]. « "En général, ils analysent les réseaux situés à deux degrés de séparation de la cible." Autrement dit, la NSA espionne aussi ceux qui communiquent avec ceux qui communiquent avec ceux qui sont espionnés » [15]. D'autres agences de renseignement européennes, comme

9. Audenard, 2013, *Bornes wifi et smartphones dans les magasins*, blogs/sécurité, Orange Business [http://www.orange-business.com/fr/blogs/securite/mobilite/souriez-vous-etes-pistes-merci-aux-bornes-wifi-des-magasins].

10. Plusieurs versions du guide publié par Facebook ont fuité [http://publicintelligence.net/facebook-law-enforcement-subpoena-guides/] ces dernières années. On trouve également plusieurs autres guides du même acabit (mais tout n'est pas forcément juste) sur cryptome.org [http://cryptome.org/isp-spy/online-spying.htm] (liens en anglais).

11. Begeek, 2013, *Facebook publie son premier rapport international des demandes gouvernementales* [http://www.begeek.fr/%20facebook-publie-premier-rapport-international-demandes-gouvernementales-102351].

12. Bruce Schneier, cité par Guillaud, 2013, *Lutter contre la surveillance : armer les contre-pouvoirs*, Internet Actu [http://www.internetactu.net/2013/06/13/lutter-contre-la-surveillance-armer-les-contre-pouvoirs/]

13. Maxime Vaudano, 2013, *Plongée dans la « pieuvre » de la cybersurveillance de la NSA*, Le Monde.fr [http://www.lemonde.fr/technologies/visuel/2013/08/27/plongee-dans-la-pieuvre-de-la-cybersurveillance-de-la-nsa_3467057_651865.html]

14. Pisani, 2007, *Facebook/5 : la recette*, Transnets [http://pisani.blog.lemonde.fr/2007/06/19/facebook5-la-recette/]

15. Manach, 2013, *Pourquoi la NSA espionne aussi votre papa (#oupas)*, Bug Brother [http://bugbrother.blog.lemonde.fr/2013/06/30/pourquoi-la-nsa-espionne-aussi-votre-papa-oupas/]

la DGSE française [16], auraient des pratiques similaires, bien que disposant de moins de moyens.

3.2 Journaux et rétention de données

La plupart des organisations qui fournissent des services sur Internet (connexion, hébergement de site, *etc.*) conservent plus ou moins de traces de ce qui transite entre leurs mains, sous forme de journaux de connexion : qui a fait quoi, à quel moment.
tome 1 § 2.4
On appelle ces journaux des « logs ».

Historiquement, ces journaux répondent à un besoin technique : ils sont utilisés par les personnes qui s'occupent de la maintenance des serveurs afin de diagnostiquer et résoudre les problèmes. Cependant, ils peuvent aussi être très utiles pour recueillir des données sur les utilisateurs de ces serveurs.

3.2.1 Lois de rétention de données

Désormais, dans la plupart des pays occidentaux, les fournisseurs de services Internet sont légalement tenus de conserver leurs journaux pendant un certain temps, pour pouvoir répondre à des requêtes légales. Les lois qui règlementent la rétention de données définissent de façon plus ou moins claire les informations qui doivent être conservées dans ces journaux. La notion de fournisseur de service Internet peut ainsi être entendue de façon assez large [17] : un cybercafé est un fournisseur de service Internet qui fournit *aussi* une machine pour accéder au réseau.

Au-delà des obligations légales, il est probable que de nombreux fournisseurs de services Internet conservent de plus ou moins grandes quantités d'information sur les internautes qui utilisent leurs services, notamment pour la publicité ciblée. Comme vu précedemment, certaines entreprises, telles Google, Yahoo ou Facebook, sont particulièrement connues pour cela. Cependant, étant donné que ce « modèle de fourniture de services adossés à de la publicité est quasiment devenu la norme » [18], on peut supposer que nombre d'autres font de même plus discrètement.

Au Royaume-Uni, un FAI a ainsi créé une polémique lorsqu'il est apparu qu'il gardait la trace de l'ensemble des pages web visitées par ses abonnés pour tester une technologie de profilage destinée à « offrir » de la « publicité comportementale » [19] [20].

Le serveur qui héberge le contenu utilisé (page web, boîte mail…) et le fournisseur d'accès à Internet sont particulièrement bien placés pour disposer des informations permettant d'identifier qui est à l'origine d'une requête de connexion. En France, ce sont eux qui sont particulièrement visés par les lois de rétention de données.

3.2.2 Les journaux conservés par les hébergeurs

On a vu que le serveur qui héberge un service (comme un site web, une boîte mail ou un salon de messagerie instantanée) avait accès à une grande quantité de données.
page 29

16. Manach, 2010, *Frenchelon : la DGSE est en « 1ère division »*, Bug Brother [http://bugbrother.blog.lemonde.fr/2010/10/02/frenchelon-la-dgse-est-en-1ere-division/]
17. CNIL, 2010, *Conservation des données de trafic : hot-spots wi-fi, cybercafés, employeurs, quelles obligations ?* [http://www.cnil.fr/documentation/fiches-pratiques/fiche/article/conservation-des-donnees-de-trafic/].
18. CNIL, *La publicité ciblée en ligne* (op. cit.), p. 4
19. CNIL, *La publicité ciblée en ligne* (op. cit.), p. 17
20. Arnaud Devillard, 2009, *Affaire Phorm : Bruxelles demande des comptes au Royaume-Uni* [http://www.01net.com/editorial/501173/affaire-phorm-bruxelles-demande-des-comptes-au-royaume-uni/]

En France, c'est la Loi pour la Confiance dans l'Économie Numérique [21] (issue de la directive européenne 2006/24/EC sur la rétention de données [22]) qui oblige les hébergeurs de contenus publics à conserver « les données de nature à permettre l'identification » de « toute personne ayant contribué à la création d'un contenu mis en ligne » [23] : écrire sur un blog ou sur un site de média participatif, envoyer un email, poster sur une liste de diffusion publique, par exemple. Concrètement, il s'agit de conserver pendant un an les éventuels identifiants ou pseudonymes fournis par l'auteur, mais surtout l'adresse IP associée à chaque modification de contenu [24]. Une page 12 requête auprès du fournisseur d'accès à Internet qui fournit cette adresse IP permet ensuite généralement de remonter jusqu'au propriétaire de la connexion utilisée.

De plus, la Loi relative à la programmation militaire [25], promulguée fin décembre 2013, permet de demander ces mêmes informations, en temps réel, pour des motifs aussi variés que : les attaques terroristes, les cyber-attaques, les atteintes au potentiel scientifique et technique, la criminalité organisée, *etc.*

C'est donc cette obligation de rétention de données qui permet à la police, dans notre histoire introductive, d'obtenir des informations auprès des organismes hébergeant les adresses emails incriminées :

> – *On va demander à Gmail ainsi qu'à no-log les informations sur ces adresses email. À partir de là on pourra sans doute mettre la main sur les personnes responsables de cette publication.*

Les hébergeurs pourront être plus ou moins coopératifs sur la façon de vérifier la légalité des requêtes que leur adressent les flics et d'y répondre : il semblerait que certains répondent à un simple email des flics alors que d'autres attendront d'avoir un courrier signé d'un juge [26], voire ne répondent pas aux requêtes [27].

Non seulement les personnes ayant accès au serveur peuvent collaborer avec les flics de plein gré, mais un adversaire peut aussi, comme dans le cas d'un ordinateur personnel, s'y introduire et espionner ce qui s'y passe en utilisant des failles, sans passer par

21. « LOI n° 2004-575 du 21 juin 2004 pour la confiance dans l'économie numérique », Journal Officiel n° 143 du 22 juin 2004 page 11168, NOR : ECOX0200175L [http://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000000801164&dateTexte=&categorieLien=id]

22. EUR-lex, 2006, *Directive 2006/24/CE du Parlement européen et du Conseil du 15 mars 2006 sur la conservation de données générées ou traitées dans le cadre de la fourniture de services de communications électroniques accessibles au public ou de réseaux publics de communications* [http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32006L0024:FR:HTML].

23. legifrance, 2011, *Décret n° 2011-219 du 25 février 2011 relatif à la conservation et à la communication des données permettant d'identifier toute personne ayant contribué à la création d'un contenu mis en ligne* [http://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000023646013&dateTexte=&categorieLien=id].

24. « Les personnes physiques ou morales qui assurent, même à titre gratuit, pour mise à disposition du public par des services de communication au public en ligne, le stockage de signaux, d'écrits, d'images, de sons ou de messages de toute nature fournis par des destinataires de ces services » (LCEN, op. cit.), c'est-à-dire les hébergeurs sont tenus de conserver pendant un an : « a) L'identifiant de la connexion à l'origine de la communication ; b) L'identifiant attribué par le système d'information au contenu, objet de l'opération ; c) Les types de protocoles utilisés pour la connexion au service et pour le transfert des contenus ; d) La nature de l'opération ; e) Les date et heure de l'opération ; f) L'identifiant utilisé par l'auteur de l'opération lorsque celui-ci l'a fourni ; » (Décret n° 2011-219 du 25 février 2011, op. cit.)

25. Legifrance, 2014, *LOI n° 2013-1168 du 18 décembre 2013 relative à la programmation militaire pour les années 2014 à 2019 et portant diverses dispositions concernant la défense et la sécurité nationale* [http://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000028338825&dateTexte&categorieLien=id].

26. Globenet, 2014, *No-log, les logs et la loi* [http://www.globenet.org/No-log-les-logs-et-la-loi.html].

27. « Précisons que les serveurs hébergeant les sites du réseau Indymedia, domiciliés aux USA à Seattle, refusent systématiquement de donner connaissance aux autorités des logs de connexion des ordinateurs consultant ces sites ou y déposant une contribution, rendant de fait non-identifiable les auteurs des contributions » (dossier d'instruction judiciaire cité par Anonymes, 2010, *Analyse d'un dossier d'instruction antiterroriste* [http://infokiosques.net/lire.php?id_article=789].

l'étape requête légale. Il aura alors accès à toutes les données stockées sur le serveur, y compris les journaux.

Mais le serveur ne connaît pas toujours l'identité réelle des clients qui s'y connectent : en général, tout ce qu'il peut donner c'est une adresse IP.

C'est alors qu'intervient le fournisseur d'accès à Internet.

3.2.3 Les journaux conservés par les fournisseurs d'accès Internet (FAI)

page 15

On a vu qu'on accédait à Internet par l'intermédiaire d'un fournisseur d'accès à Internet (FAI). Ce FAI est en général une société qui fournit une « box » connectée à Internet. Mais ça peut aussi être une association ou une institution publique (une université, par exemple, quand on utilise leurs salles informatiques). Les FAI sont eux aussi soumis à des lois concernant la rétention de données.

Au sein de l'Union Européenne, une directive oblige les fournisseurs d'accès à Internet (FAI) à garder la trace de qui s'est connecté, quand, et depuis où. En pratique, cela consiste à enregistrer quelle adresse IP a été assignée à quel abonné pour quelle période [28]. Les institutions qui fournissent un accès à Internet, comme les bibliothèques et les universités, font de même : en général il faut se connecter avec un nom d'utilisateur et un mot de passe. On peut ainsi savoir qui utilisait quel poste à quel moment. La directive européenne précise que ces données doivent être conservées de 6 mois à 2 ans. En France, la durée légale est de un an [29].

De plus, FAI et hébergeurs français sont tenus de conserver les « informations fournies lors de la souscription d'un contrat par un utilisateur ou lors de la création d'un compte » pendant un an après la fermeture du compte. « Lorsque la souscription du contrat ou du compte est payante », ils doivent aussi conserver les informations relatives au paiement [30].

L'objectif des lois de rétention de données est donc de rendre facile, pour les autorités, d'associer un nom à tout geste effectué sur Internet.

Des flics qui enquêteraient par exemple sur un article publié sur un blog peuvent demander aux responsables du serveur hébergeant le blog l'adresse IP de la personne qui a posté l'article, ainsi que la date et l'heure correspondantes. Une fois ces informations obtenues, ils vont demander au fournisseur d'accès à Internet responsable de cette adresse IP à qui elle était assignée au moment des faits.

---

28. « Les personnes dont l'activité est d'offrir un accès à des services de communication au public en ligne »(LCEN, op. cit.), c'est-à-dire les FAI, sont tenues de conserver durant un an : « a) L'identifiant de la connexion ; b) L'identifiant attribué par ces personnes à l'abonné ; c) L'identifiant du terminal utilisé pour la connexion lorsqu'elles y ont accès ; d) Les dates et heure de début et de fin de la connexion ; e) Les caractéristiques de la ligne de l'abonné ; » (Décret n° 2011-219 du 25 février 2011, op. cit.)

29. Parlement Européen et Conseil, 2006, *Directive 2006/24/CE du Parlement européen et du Conseil du 15 mars 2006 sur la conservation de données générées ou traitées dans le cadre de la fourniture de services de communications électroniques accessibles au public ou de réseaux publics de communications, et modifiant la directive 2002/58/CE* [http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32006L0024:FR:HTML]

30. « 3° Pour les personnes mentionnées aux 1 et 2 du I du même article [FAI et hébergeurs, NdA], les informations fournies lors de la souscription d'un contrat par un utilisateur ou lors de la création d'un compte : a) Au moment de la création du compte, l'identifiant de cette connexion ; b) Les nom et prénom ou la raison sociale ; c) Les adresses postales associées ; d) Les pseudonymes utilisés ; e) Les adresses de courrier électronique ou de compte associées ; f) Les numéros de téléphone ; g) Le mot de passe ainsi que les données permettant de le vérifier ou de le modifier, dans leur dernière version mise à jour ; 4° Pour les personnes mentionnées aux 1 et 2 du I du même article [FAI et hébergeurs, NdA], lorsque la souscription du contrat ou du compte est payante, les informations suivantes relatives au paiement, pour chaque opération de paiement : a) Le type de paiement utilisé ; b) La référence du paiement ; c) Le montant ; d) La date et l'heure de la transaction. Les données mentionnées aux 3° et 4° ne doivent être conservées que dans la mesure où les personnes les collectent habituellement. » (Décret n° 2011-219 du 25 février 2011, op. cit.)

– *Quelle histoire ! Mais quel rapport avec nos bureaux ?*
– *Et bien c'est également pour ça que je vous appelle. Ils affirment qu'ils ont toutes les preuves comme quoi ces documents ont été publiés depuis vos bureaux. Je leur ai dit que ce n'était pas moi, que je ne voyais pas de quoi ils parlaient.*

C'est exactement de ça qu'il s'agit quand, dans notre histoire du début, la police prétend, traces à l'appui, que les relevés bancaires ont été postés depuis les bureaux rue Jaurès. Elle a au préalable obtenu auprès des hébergeurs du site de publication l'adresse IP qui correspond à la connexion responsable de la publication des documents incriminés. Cette première étape permet de savoir d'où, de quelle « box », provient la connexion. La requête auprès du FAI permet de savoir quel est le nom de l'abonné – adresse en prime – *via* son contrat, associé à l'adresse IP.

3.2.4 Requêtes légales

En France, lorsque les flics souhaitent accéder aux journaux prévus par les lois de rétention de données, ils sont supposés passer par une *requête légale* : une demande officielle qui oblige les personnes qui administrent un serveur à leur fournir les informations demandées… ou à désobéir. Ces requêtes légales sont supposées préciser les informations demandées. Mais elles ne le font pas toujours, et les fournisseurs de services Internet donnent parfois davantage d'informations que ce que la loi les oblige à fournir.

Voici l'extrait d'une requête légale reçue par un hébergeur de mail français, l'adresse mail du compte visé a été anonymisée en remplaçant l'identifiant par *adresse*. L'orthographe n'a pas été modifiée.

REQUISITION JUDICIAIRE

Lieutenant de Police En fonction à la B.R.D.P

Prions et, au besoin, requérons :

Monsieur le président de l'association GLOBENET 21ter, rue Voltaire 75011 Paris

à l'effet de bien vouloir :

Concernant l'adresse de messagerie adresse@no-log.org

- Nous communiquer **l'identité complète** (nom, prénom date de naissance, filiation) et les **coordonnées** (postales, téléphoniques, électroniques et bancaires) de son **titulaire**
- Nous indiquer les TRENTE dernières données de connexion (adresse IP, date heures et fuseau horaire) utilisées pour **consulter, relever où envoyer des messages** avec ladite adresse (Pop, Imap ou Webmail)
- Nous indiquer si une **redirection est active** sur cette messagerie, et nous communiquer le ou les e-mails de destination, le cas échéant
- Nous communiquer le **numéro de téléphone** à l'abonnement internet du compte no-log.org « adresse » et les **trente dernières données de connexion** qui lui sont relatives
- Nous communiquer les TRENTES **dernières données de connexion** (adresse IP, date heure et fuseau horaire) aux **pages d'aministration** du compte no-log « adresse »

De plus, il est avéré que les flics demandent parfois de telles informations dans un simple courrier électronique, et il est probable que de nombreux fournisseurs de services Internet répondent directement à de telles requêtes officieuses, ce qui implique

que *n'importe qui* peut obtenir de telles informations en se faisant passer pour la police.

Les requêtes légales sont monnaie courante. Les gros fournisseurs de services Internet ont désormais des services légaux dédiés pour y répondre, et une grille tarifaire chiffre chaque type de demande [31]. Depuis octobre 2013, en France, une grille tarifaire indexée par l'État vient même homogénéiser ces différentes prestations [32] : identifier un abonné à partir de son adresse IP coûtait ainsi 4 € (tarifs en vigueur en octobre 2013). Au-delà de 20 demandes, ce tarif est réduit à 18 centimes.

Ainsi pour la seconde moitié de l'année 2013, Google a reçu chaque mois, en moyenne, 458 demandes de renseignements sur ses utilisateurs de la part de la France, concernant au total 3378 comptes – des chiffres en augmentation constante depuis 2009. Après analyse de la recevabilité des requêtes sur le plan juridique, la société a répondu à 51% d'entre elles [33] : l'autre moitié des requêtes n'entrait donc pas dans le cadre de ce que l'entreprise s'estimait légalement contrainte de fournir.

3.3 Écoutes de masse

Au-delà des journaux et des requêtes légales prévus par les lois de rétention de données, les communications sur Internet sont surveillées de façon systématique par divers services étatiques.

Un ancien employé de l'opérateur de télécommunications états-unien AT&T a témoigné [34] du fait que la NSA (l'agence de renseignement électronique états-unienne) surveillait l'ensemble des communications Internet et téléphoniques qui passaient par une importante installation de l'opérateur de télécommunication AT&T à San Francisco. Ceci, à l'aide d'un superordinateur spécialement conçu pour la surveillance de masse, en temps réel, de communications [35]. Il a aussi déclaré que de telles installations existaient probablement au sein d'autres infrastructures similaires dans d'autres villes des États-Unis, ce que confirment les révélation d'un ex-employé de la NSA et de la CIA [36]. Des installations similaires seraient mises en place par les services secrets britanniques sur plus de 200 fibres optiques sous-marines [37].

La NSA a aussi obtenu un accès direct aux serveurs de plusieurs « géants » du net (Microsoft, Yahoo, Google, Facebook, PalTalk, Youtube, Skype, AOL et Apple) [38], ce qui lui permet d'accéder aux données qu'ils hébergent ou qui transitent par leurs serveurs [39].

---

31. Christopher Soghoian, 2010, *Your ISP and the Government : Best Friends Forever* [http://www.defcon.org/html/defcon-18/dc-18-speakers.html#Soghoian] (en anglais).
32. legifrance, 2013, *Arrêté du 21 août 2013 pris en application des articles R. 213-1 et R. 213-2 du code de procédure pénale fixant la tarification applicable aux réquisitions des opérateurs de communications électroniques* [http://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000028051025].
33. Google, 2014, *France - Google Transparency Report* [https://www.google.com/transparencyreport/userdatarequests/FR/].
34. Mark Klein, 2004, *AT&T's Implementation of NSA Spying on American Citizens* [http://www.audioactivism.org/text/att_klein_wired.pdf] (en anglais).
35. Reflets.info, 2011, *#OpSyria : BlueCoat maître artisan de la censure syrienne* [http://reflets.info/opsyria-bluecoat-maitre-artisan-de-la-censure-syrienne/].
36. Craig Timberg et Barton Gellman, 2013, *NSA paying U.S. companies for access to communications networks* [http://www.washingtonpost.com/world/national-security/nsa-paying-us-companies-for-access-to-communications-networks/2013/08/29/5641a4b6-10c2-11e3-bdf6-e4fc677d94a1_story.html] (en anglais).
37. L'expansion.com, 2013, *"Operation Tempora" : comment les Britanniques dépassent les Américains pour espionner Internet* [http://lexpansion.lexpress.fr/high-tech/operation-tempora-comment-les-britanniques-depassent-les-americains-pour-espionner-internet_390971.html].
38. NSA, 2013, *Dates When PRISM Collection Began For Each Provider* [https://commons.wikimedia.org/wiki/File:Prism_slide_5.jpg].
39. Le Monde, 2013, *Le FBI aurait accès aux serveurs de Google, Facebook, Microsoft, Yahoo ! et d'autres géants d'Internet* [http://www.lemonde.fr/ameriques/article/2013/06/07/le-fbi-a-acces-aux-serveurs-des-geants-d-internet_3425810_3222.html].

De même, les communications satellites sont écoutées par le réseau Echelon, un « système mondial d'interception des communications privées et publiques » [40] élaboré par des pays anglo-saxons [41]. Les informations à ce sujet restent floues, mais la France semble aussi disposer d'un réseau d'écoute des télécommunications sur son territoire [42].

La NSA surveille et recoupe également les échanges d'emails pour établir une carte des relations entre tous les habitants des États-Unis [43]. Si ce genre de pratiques n'est pas forcément attesté ailleurs dans le monde, elles y sont tout aussi possibles.

De plus, pour toute organisation ayant les moyens d'être un nœud conséquent du réseau, que ce soit officiellement ou non, l'utilisation du *Deep Packet Inspection* (ou *DPI* : Inspection en profondeur des paquets, en français) se généralise. L'avantage de cette technique par rapport aux techniques classiques est que la surveillance ne se limite plus aux seules informations inscrites dans les en-têtes des paquets IP, mais touche au contenu même des communications : si celles-ci ne sont pas chiffrées, il est possible de retrouver par exemple le contenu complet d'emails, ou l'intégralité de nos consultations et recherches sur le web.

page 28
page 28

L'utilisation de cette technique, en Lybie ou en Syrie par exemple, a permis dans un premier temps de mettre sous surveillance numérique toute la population du pays, pour dans un second temps notamment effectuer des attaques ciblées. La société Amesys, basée en France, a en effet, avec l'aide et l'appui du gouvernement [44] de l'époque, installé de tels systèmes en Lybie [45], au Maroc, au Qatar [46] ou encore en France [47].

plus bas

3.4 Attaques ciblées

Lorsqu'une internaute ou qu'une ressource disponible *via* Internet – comme un site web ou une boîte mail – éveille la curiosité d'un adversaire, ce dernier peut mettre en place des attaques ciblées. Ces attaques ciblées peuvent avoir lieu à différents niveaux : les annuaires qui permettent de trouver la ressource, les serveurs qui l'hébergent, les clients qui y accèdent, *etc.* Nous étudions ces différentes possibilités dans cette partie.

En France, la loi oblige les fournisseurs d'accès à Internet à bloquer l'accès aux sites web qui ont été inscrits sur une « liste noire » à la suite d'une décision de justice [48].

C'est ainsi qu'en octobre 2011, le tribunal de Grande Instance de Paris a ordonné à sept fournisseurs d'accès à Internet français de bloquer « par IP ou par DNS » le site web https://copwatchnord-idf.org/ [49] ; ce site était accusé de propos injurieux et diffamatoires, et de collecter des données à caractère personnel sur des policiers.

40. Wikipédia, 2014, *Echelon* [https://fr.wikipedia.org/wiki/Echelon].
41. Gerhard Schmid, 2001, *Rapport sur l'existence d'un système d'interception mondial des communications privées et économiques (système d'interception ECHELON)* [http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-//EP//NONSGML+REPORT+A5-2001-0264+0+DOC+PDF+V0//FR].
42. Wikipédia, 2014, *Frenchelon* [https://fr.wikipedia.org/wiki/Frenchelon].
43. Gorman, Siobhan, 2008, *NSA's Domestic Spying Grows As Agency Sweeps Up Data : Terror Fight Blurs Line Over Domain; Tracking Email.* [http://online.wsj.com/article/SB120511973377523845.html] (en anglais).
44. kitetoa, 2011, *Amesys : le gouvernement (schizophrène) français a validé l'exportation vers la Libye de matériel d'écoute massive des individus*, Reflets.info [http://reflets.info/amesys-le-gouvernement-schizophrene-francais-a-valide-lexportation-vers-la-libye-de-materiel-decoute-massive-des-individus/].
45. Fabrice Epelboin, 2011, *Kadhafi espionnait sa population avec l'aide de la France* [http://reflets.info/kadhafi-espionnait-sa-population-avec-l%E2%80%99aide-de-la-france/].
46. Reflets.info, 2011, *Qatar : Le Finger tendu bien haut d'Amesys* [http://reflets.info/qatar-le-finger-tendu-bien-haut-damesys/].
47. Jean Marc Manach, 2011, *Amesys surveille aussi la France* [http://owni.fr/2011/10/18/amesys-surveille-france-takieddine-libye-eagle-dga-dgse-bull/].
48. legifrance, 2011, *Article 4 de la loi LOPPSI 2* [http://www.legifrance.gouv.fr/affichTexteArticle.do?idArticle=JORFARTI000023707337&cidTexte=JORFTEXT000023707312&dateTexte=29990101].
49. Tribunal de Grande Instance de Paris, 2011, *jugement en référé du 14 octobre 2011 ordonnant le blocage Copwatch* [http://www.pcinpact.com/media/20111014_tgi_paris_copwatch.pdf].

En février 2012, le tribunal ordonnait le blocage de l'un des 35 sites miroirs [50] que le ministère de l'Intérieur voulait faire bloquer [51].

Par contre, le tribunal n'a pas ordonné le blocage des 34 autres miroirs référencés par le ministère de l'Intérieur, car ce dernier « n'indique pas s'il a tenté ou non d'identifier leurs éditeurs et leurs hébergeurs », ni celui des sites miroirs qui pourraient apparaître.

3.4.1 Bloquer l'accès au fournisseur de ressources

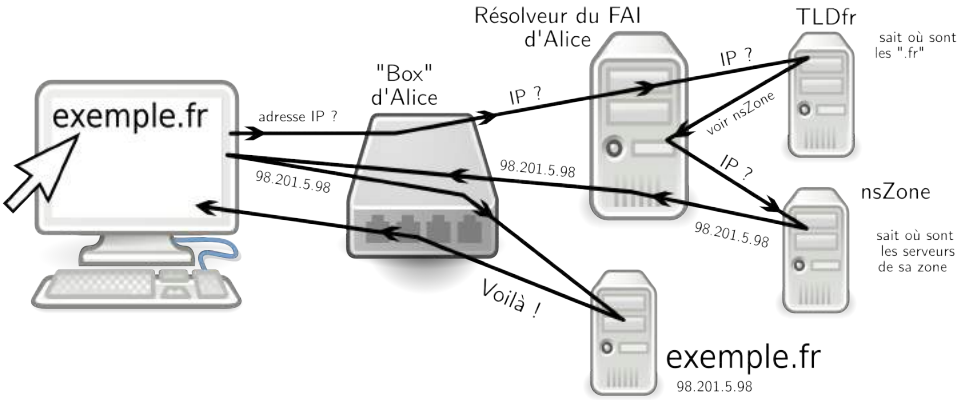
Penchons-nous maintenant sur les différents moyens qui permettent de bloquer l'accès à une ressource sur Internet.

Saisie de domaines

Il est possible de détourner le trafic qui devait aller vers un certain nom d'hôte en modifiant l'annuaire utilisé pour passer du nom de domaine à l'adresse IP, c'est-à-dire le DNS.

page 21

Cela peut se faire à différents niveaux.



Les étapes clefs d'une requête DNS

Pour des raisons d'efficacité et de robustesse, le Domain Name Système est géré par diverses organisations, en un système d'information hiérarchisé et distribué.

La base de données globale du DNS est répartie entre plusieurs serveurs de noms, chacun de ces serveurs ne maintenant qu'une partie de la base. Ainsi tous les domaines finissant par .fr relèvent du serveur de nom de l 'AFNIC, une association créée à cet effet en 1997. De même, c'est une Société Anonyme étasunienne cotée en bourse, Verisign, qui a reçu la délégation pour caractériser l'emplacement (l'adresse IP) de tous les domaines finissant par .com ou l'emplacement de l'organisation à qui Verisign a elle-même délégué une zone à gérer.

On peut lire la liste des organisations et entreprises qui sont chargées de gérer les noms dits *de premier niveau* (TLD, Top Level Domain) comme .com, .fr, .org, *etc.* sur le site web de l'IANA [52] (Internet Assigned Numbers Authority), qui gère le serveur racine du DNS, celui qui fait autorité sur tous les autres.

Si les gestionnaires au niveau des TLD ont un rôle purement technique (tenir à jour une liste des domaines dont ils ont la charge), ceux à qui elle délègue sont généralement des entreprises commmerciales (appelées *registrars*) qui vendent des noms de domaine.

50. Un site miroir est une copie exacte d'un autre site web.
51. Tribunal de Grande Instance de Paris, 2012, *ordonnance de référé rendue le 10 février 2012* [http://cnd.pcinpact.com/media/20120210_tgi_paris_ordonnance_refere_copwatch_v2.pdf].
52. IANA, 2014, *Root Zone Database* [http://www.iana.org/domains/root/db] (en anglais).

Ainsi, acheter (ou louer) un nom de domaine est une opération disctincte de louer une IP : par exemple, pour monter son propre site web, il faudra d'une part acheter un nom de domaine et d'autre part trouver un hébergement pour le site, avec une adresse IP qui lui est attachée. Et ensuite mettre en place la liaison entre les deux. Certaines entreprises proposent tous ces services en même temps, mais ce n'est ni systématique ni obligatoire.

On voit maintenant se dessiner une carte des points névralgiques où peut intervenir la censure.

La saisie de nom de domaine la plus spectaculaire à ce jour fut certainement celle inscrite dans le cadre de la fermeture du site d'hébergement de fichiers megaupload.com par le Département de la Justice des États-Unis. Pour rendre inaccessibles les services de ce site, le FBI a notamment demandé à Verisign, l'entreprise qui gère les *.com*, de modifier ses tables de correspondance afin que cette adresse pointe non plus vers les serveurs de Megaupload mais vers un serveur du FBI indiquant que le site avait été saisi [53].

Cependant, une des premières censures connues par suspension d'un nom de domaine s'est produite, en 2007, au niveau d'un registrar : GoDaddy (le plus important au monde). Dans le cadre d'un conflit entre un de ses clients, seclists.org, et un autre site, myspace.com, GoDaddy prit le parti de ce dernier et modifia sa base de données, rendant, du jour au lendemain et sans avertir personne, le site injoignable [54] (sauf pour les personnes connaissant son adresse IP par cœur).

Enfin, si modifier les annuaires globaux n'est à la portée que de quelques États et sociétés, nombreux sont ceux qui peuvent simplement falsifier leur propre version de l'annuaire.

Ainsi, chaque fournisseur d'accès à Internet (FAI) a en général ses propres serveurs de noms de domaines, qui sont utilisés par défaut par ses abonnés. Début 2013, le FAI Free a utilisé un résolveur intégré à la freebox pour mettre en œuvre son nouveau service de blocage de publicités [55].

À moins de désactiver ce service ou de configurer son ordinateur pour ne plus utiliser les DNS par défaut de Free, ce résolveur, pour une liste de nom de domaine à bloquer, donne une réponse "fausse", en pointant vers un serveur de fichier vide et par conséquent, le navigateur web n'affiche rien à la place des pub.

Hameçonnage

Dans le même ordre d'idée, l'hameçonnage [56] (appelé également filoutage, ou *phishing* en anglais) consiste à pousser l'internaute à se connecter à un site qui n'est pas celui qu'il croit être, mais qui y ressemble beaucoup. Par exemple, un site qui ressemble comme deux gouttes d'eau à celui d'une banque, afin d'obtenir des mots de passe de connexion à une interface de gestion de comptes bancaires. Pour cela, l'adversaire achète un nom de domaine qu'on croira être le bon au premier coup d'œil. Il ne lui reste plus qu'à inciter la personne ciblée à se connecter à ce site, généralement en lui faisant peur, par exemple « Nous avons détecté une attaque sur votre compte » ou « Vous avez dépassé votre quota », suit alors la proposition de régulariser la situation en cliquant sur le lien piégé.

53. Après cette coupure, des milliers d'utilisateurs se sont vus privés de leurs contenus en un claquement de doigts (et pas que de leurs fichiers pirates, au vu des pétitions en ligne et de tous ces gens disant que leur vie professionnelle était ruinée car ils n'avaient plus accès à tous leurs documents).

54. Fyodor, 2007, *Seclists.org shut down by Myspace and GoDaddy* [http://seclists.org/nmap-announce/2007/0] (en anglais).

55. S. Bortzmeyer, 2013, *Comment Free bloque les pubs* [http://www.bortzmeyer.org/free-adgate.html].

56. Voir à ce sujet Wikipédia, 2014, *Hameçonnage* [https://fr.wikipedia.org/wiki/Hameçonnage], qui explique notamment quelques parades (partielles) à cette attaque.

Pour que le nom de domaine affiché ressemble lui aussi comme deux gouttes d'eau à celui du site copié, il existe plein de techniques : l'adversaire peut par exemple utiliser des caractères spéciaux qui ont l'apparence des caractères de l'alphabet latin. Ainsi, en substituant un « e » cyrillique à un « e » latin dans `exemple.org`, on obtient une adresse qui s'affiche de façon (quasi) identique à l'originale, mais qui représente pour l'ordinateur une adresse différente ; on trouve parfois aussi des tirets en plus ou en moins (ma-banque.fr au lieu de mabanque.fr) ; il s'agit parfois d'un nom identique, avec un nom de domaine de premier niveau (*top-level domain*, ou TDL : .com, .net, .org, .fr…) différent (site.com au lieu de site.org) ; certains utilisent aussi des sous-domaines (paypal.phishing.com renvoie vers le site de phishing, et non vers paypal.com), *etc.*

Une parade intégrée dans les navigateurs web consiste à avertir l'utilisateur du danger et à lui demander une confirmation avant d'accéder au site suspect. Cela dit, cette solution nécessite que le navigateur web contacte une base de données centralisée, recensant les sites considérés comme malveillants, et peut donc poser des problèmes de discrétion : le serveur hébergeant cette liste aura nécessairement connaissance des sites que l'on visite.

3.4.2 Attaquer le serveur

Une autre catégorie d'attaques consiste, pour l'adversaire, à s'en prendre à l'ordinateur qui héberge la ressource qui l'intéresse. Ça peut se faire physiquement ou à distance.

Saisie de serveurs

Il s'agit tout simplement pour un adversaire qui en a les moyens, par exemple la police ou la justice, d'aller là où se trouve l'ordinateur auquel il s'intéresse. L'adversaire peut alors s'emparer de la machine, ou copier les données qu'elle abrite. Il pourra ensuite étudier toutes les traces qui ont été laissées dessus par les personnes qui s'y sont connectées… du moins si son disque dur n'est pas chiffré.

tome 1 ch. 2

tome 1 § 5.1

Au moins quatorze serveurs ont été saisis par la justice en Europe entre 1995 et 2007 [57]. Ainsi en 2007, un serveur de Greenpeace Belgique a été emmené par la police belge suite à une plainte pour « association de malfaiteurs » d'une compagnie d'électricité belge [58] contre laquelle l'organisation écologiste avait appelé à manifester.

Piratage de serveurs

Comme tout ordinateur, un serveur peut être *piraté* : cela consiste pour l'attaquant à s'introduire « par effraction » dans l'ordinateur. Des erreurs de conception ou de programmation, qui permettent de détourner le fonctionnement d'un programme et de s'introduire dans l'ordinateur sur lequel il fonctionne, sont régulièrement découvertes dans les programmes couramment utilisés sur les serveurs. Des erreurs dans la configuration des logiciels de la part des admins de ces serveurs sont aussi possibles.

Ainsi, en avril 2011, l'exploitation de failles dans les logiciels utilisés sur leurs serveurs a permis à des pirates de s'introduire dans les serveurs de Sony Online, du PlayStation Network et de Qriocity (Sony Entertainment Network). Cela leur a donné accès aux données personnelles et bancaires de millions d'utilisateurs de ces réseaux de jeux vidéo [59] : pseudonymes, mots de passe, adresses postales et électroniques, *etc.*

57. Globenet, 2007, *Les saisies de serveurs en Europe : un historique* [http://www.globenet.org/Les-saisies-de-serveurs-en-Europe.html?start_aff=6].

58. Greenpeace, 2007, *Greenpeace déplore l'abus de procédure et la réaction disproportionnée d'Electrabel* [http://www.greenpeace.org/belgium/fr/pers/persberichten/perquisition/].

59. Wikipédia, 2014, *Piratage du PlayStation Network* [https://fr.wikipedia.org/wiki/Piratage_du_PlayStation_Network] ; Diowan, 2011, *Retour sur le piratage de Sony* [http://www.jeuxvideo.com/dossiers/00014882/le-piratage-du-psn.htm]

Si cet exemple a beaucoup fait parler de lui, les failles qui rendent ce genre de piratage possible ne sont pas rares, et n'importe quel serveur peut être touché. Une fois introduits dans le serveur, les pirates peuvent potentiellement avoir accès à distance à toutes les données enregistrées sur celui-ci.

Attaque par déni de service

Sans saisir le serveur ni même le pirater, il est possible d'empêcher celui-ci de fonctionner en le saturant : l'adversaire fait en sorte que de très nombreux robots tentent en permanence de se connecter au site à attaquer. Au delà d'un certain nombre de requêtes, le logiciel serveur est débordé et n'arrive plus à répondre : le site est alors inaccessible. On appelle cette attaque une *attaque par déni de service*[60]. Les robots utilisés pour ce type d'attaque sont souvent des logiciels malveillants installés sur des ordinateurs personnels à l'insu de leurs propriétaires.

page 22
tome 1 § 3.1

3.4.3 Sur le trajet

Enfin, un adversaire qui contrôle une partie du réseau – comme un fournisseur d'accès à Internet – peut écouter ou détourner des paquets de plusieurs manières.

Filtrage

Comme évoqué précédemment, un adversaire qui contrôle l'un des routeurs par lesquels passe le trafic entre un internaute et une ressource peut lire plus ou moins en profondeur le contenu des paquets et éventuellement le modifier, et ce d'autant plus facilement s'il n'est pas chiffré.

page 28
page 16

De nos jours, quasiment tous les fournisseurs d'accès à Internet pratiquent ce genre d'inspection, le *DPI*, a minima à des fins de statistiques. De plus, ils sont de plus en plus nombreux, de façon plus ou moins discrète, plus ou moins assumée, à s'en servir pour faire passer certains paquets avant les autres, en fonction de leur destination ou de l'application à laquelle ils correspondent. Par exemple pour ralentir la vidéo à la demande, qui génère beaucoup de trafic (et donc leur coûte cher), et privilégier le téléphone par Internet[61]. Ce type de moyens est par exemple utilisé par le FAI SFR[62] afin de modifier les pages web visitées par ses abonnés en 3G[63].

page 40

Le déploiement massif d'équipements permettant cet examen approfondi des paquets rend beaucoup plus facile une surveillance aux portes des réseaux des FAI.

En analysant ce type de données, les gouvernements peuvent identifier la position d'un individu, de ses relations et des membres d'un groupe, tels que « des opposants politiques. »[64]. De tels systèmes ont été vendus par des sociétés occidentales à la Tunisie, à l'Égypte, à la Libye, au Bahreïn et à la Syrie[65], et sont également en service dans certains pays occidentaux. Ceux-ci permettent, sur la base d'une surveillance de masse, de cibler des utilisateurs et filtrer, censurer du contenu.

page 40

60. Wikipédia, 2014, *Attaque par déni de service* [https://fr.wikipedia.org/wiki/Ddos].
61. Christopher Parsons et Colin Bennet, 2010, *What Is Deep Packet Inspection* [http://www.deeppacketinspection.ca/what-is-dpi/] (en anglais).
62. bluetouff, 2013, *SFR modifie le source HTML des pages que vous visitez en 3G* [http://reflets.info/sfr-modifie-le-source-html-des-pages-que-vous-visitez-en-3g/].
63. Wikipédia, 2014, *3G* [https://fr.wikipedia.org/wiki/3G].
64. Elaman, 2011, *Communications monitoring solutions* [http://wikileaks.org/spyfiles/docs/elaman/188_communications-monitoring-solutions.html] (en anglais).
65. Jean Marc Manach, 2011, *Internet massivement surveillé* [http://owni.fr/2011/12/01/spy-files-interceptions-ecoutes-wikileaks-qosmos-amesys-libye-syrie/].

Écoutes

À l'instar des bonnes vieilles écoutes téléphoniques, il est tout à fait possible d'enregistrer tout ou partie des données qui passent par un lien réseau : on parle d'« interceptions IP ». Cela permet par exemple d'écouter tout le trafic échangé par un serveur, ou celui qui passe par une connexion ADSL domestique.

Selon un article du Figaro [66], les flics français n'effectueraient « que » 500 « interceptions sur Internet » par an contre environ 5 500 écoutes téléphoniques ou interceptions de fax [67]… mais ils comptent bien rattraper leur retard dans ce domaine.

Si l'on ne prend pas de précautions particulières, une interception IP révèle à un adversaire une bonne partie de nos activités sur Internet : pages web visitées, emails et leurs contenus, conversations de messagerie instantanée… tout ce qui sort de notre ordinateur « en clair ». Le chiffrement des communications rend l'analyse du contenu issu de ces écoutes beaucoup plus difficile : l'adversaire a toujours accès aux données échangées, mais il ne peut pas les comprendre et les exploiter directement. Il peut alors essayer de casser le chiffrement utilisé… ou tenter de contourner la façon dont il est mis en œuvre. On parlera plus loin de ces questions liées au chiffrement. Dans tous les cas, l'adversaire aura toujours accès à un certain nombre d'informations précieuses, comme par exemple les adresses IP des différents interlocuteurs impliqués dans une communication.

page 59
page 12

Analyse du trafic réseau

Lorsque le trafic est chiffré, il reste possible de mettre en place des attaques plus subtiles. Un adversaire pouvant écouter le trafic réseau, même s'il n'a pas accès au contenu, dispose d'autres indices, comme la quantité d'informations transmises à un moment donné.

Ainsi, si Alice envoie 2 Mo de données chiffrées vers un site web de publication, et que quelques instants plus tard un nouveau document de 2 Mo apparaît sur ce site, cet adversaire pourra en déduire qu'il est probable que ce soit Alice qui ait envoyé ce document.

En étudiant la quantité d'informations transmises par unité de temps, l'adversaire peut aussi dessiner une « forme » : on l'appellera le *motif de données*. Le contenu d'une page web chiffrée n'aura ainsi pas le même motif qu'une conversation de messagerie instantanée chiffrée.

De plus, si un même motif de données est observé à deux points du réseau, l'adversaire peut supposer qu'il s'agit d'une même communication.

Pour prendre un exemple précis : considérons un adversaire qui écoute la connexion ADSL d'Alice, et qui observe du trafic chiffré qu'il ne peut pas déchiffrer, mais qui soupçonne Alice de discuter avec Betty par messagerie instantanée chiffrée. Considérons qu'il a également les moyens de mettre sous écoute la connexion de Betty. S'il observe une forme similaire entre les données sortant de chez Alice et celles entrant chez Betty quelques (milli)secondes plus tard, il sera conforté dans son hypothèse – sans toutefois disposer d'une preuve formelle.

Ce type d'attaque permet de confirmer une hypothèse préexistante, mais pas d'en élaborer une à partir des seules informations collectées, à moins que l'adversaire n'aie les moyens d'écouter *tout* le réseau où se situe le trafic entre Betty et Alice, et qu'il dispose d'une puissance de calcul colossale. L'existence d'un adversaire global de ce

66. Fabrice Amedeo, 2011, *La France mal armée pour enquêter sur le Net*, Le Figaro [http://www.lefigaro.fr/actualite-france/2011/04/25/01016-20110425ARTFIG00443-la-france-mal-armee-pour-enqueter-sur-le-net.php].

67. La Documentation française, 2013, *Commission nationale de contrôle des interceptions de sécurité - 21e rapport d'activité 2012-2013* [http://www.ladocumentationfrancaise.fr/var/storage/rapports-publics/144000007/0000.pdf].

type est techniquement possible, mais peu réaliste. Par contre, des agences comme la NSA sont capables de mener ce type d'attaque, au moins à l'échelle de leur pays : la NSA dispose d'une puissance de calcul qui peut être suffisante et des fuites indiquent qu'elle écouterait 75 % du trafic Internet des États-Unis d'Amérique [68].

3.4.4 Piratage du client

L'ordinateur de l'internaute, lui aussi, peut être une cible. De la même façon que dans un serveur, un attaquant peut s'introduire par effraction dans un ordinateur personnel. Des erreurs de programmation ou d'autres failles dans le système d'exploitation ou dans les applications installées permettent parfois à des adversaires d'effectuer un tel piratage – légal ou illégal – depuis Internet, sans avoir d'accès physique à la machine. De plus, l'intrusion peut être facilitée par de mauvaises pratiques de la part des utilisateurs, comme ouvrir une pièce jointe frauduleuse ou installer des programmes trouvés au hasard sur le web.

tome 1 ch. 3

Un groupe de hackers allemands renommé, le Chaos Computer Club, a mis la main sur un mouchard utilisé par la police allemande qui lui permettait d'espionner et de contrôler un ordinateur à distance [69]. De tels mouchards peuvent être installés à distance et sont aussi autorisés par la loi française dans le cadre d'une enquête sur des infractions relevant de criminalité ou de la délinquance organisée [70].

Mais « l'espionnage à distance » n'est pas seulement réservé aux pratiques policières. Aux États-Unis, c'est un lycée qui s'est lancé dans l'espionnage de grande ampleur. Sous couvert de vouloir « retrouver des ordinateurs portables volés ou perdus », le lycée avait installé une « fonction » permettant d'allumer, au bon vouloir de l'établissement, la webcam des quelques milliers d'ordinateurs distribués aux élèves. L'affaire a été révélée fin 2009 : un des élèves s'est vu reprocher d'avoir eu un « comportement inapproprié », en l'occurrence d'avoir consommé de la drogue. Le responsable accusant cet élève produisit, en guise de preuve, une photo qui s'est révélée avoir été prise à l'insu de l'étudiant, par la webcam de son ordinateur lorsqu'il était chez lui dans sa chambre ! [71]

3.5 En conclusion

Identification de l'internaute par son adresse IP, lecture de l'origine et de la destination des paquets par le biais de leurs en-têtes, enregistrement de diverses informations à différentes étapes du parcours, voire accès au contenu même des échanges… tout ceci est plus ou moins simple en fonction de l'entité impliquée.

Pirate, publicitaire, gendarme de Saint-Tropez ou NSA n'ont en effet pas les mêmes possibilités techniques et légales d'accès aux traces évoquées dans ce chapitre.

On se contentera simplement d'observer, pour conclure, que la manière dont Internet fut conçu et est le plus couramment utilisé est quasiment transparente pour un adversaire un tant soit peu attentif… à moins d'utiliser toute une série de parades adaptées pour rendre ces indiscrétions plus difficiles ; ces parades seront évoquées plus loin.

68. latribune.fr, 2012, *A peine 25% du trafic web américain échappe à la surveillance du NSA* [http://www.latribune.fr/actualites/economie/international/20130821trib000781040/a-peine-25-du-trafic-web-americain-echappe-a-la-surveillance-du-nsa.html].
69. Mark Rees, 2011, *Le CCC dissèque un cheval de Troie gouvernemental troué*, PCInpact [http://www.pcinpact.com/news/66279-loppsi-ccc-cheval-de-troie-faille-malware.htm]
70. Legifrance, 2014, *De la captation des données informatiques*, Code de procédure pénale, article 706-102-1 [http://www.legifrance.gouv.fr/affichCodeArticle.do?idArticle=LEGIARTI000023712497&cidTexte=LEGITEXT000006071154]
71. Me, myself and the Internet, 2011, *Mais qui surveillera les surveillants ?* [http://memyselfandinternet.wordpress.com/2011/02/14/«-mais-qui-surveillera-les-surveillants-»/]

Chapitre 4

Web 2.0

Le terme web 2.0 est de nos jours presque une banalité. Pour autant, il semble difficile d'en saisir la véritable consistance à force d'emploi à tort et à travers ou au contraire de définitions parfois trop techniques [1].

Il s'agit avant tout d'un terme marketing, qui définit une évolution du web à une époque où la massification de l'accès à l'Internet en fait un marché juteux. Nombre d'entreprises ne peuvent plus se permettre de l'ignorer, que leur domaine d'activité soit les médias, la communication ou le commerce. Il a bien fallu qu'elles adaptent leur « business model » à ce nouveau marché.

L'arrivée de ces nouveaux acteurs sur un web jusque-là composé principalement d'universitaires et de passionnés a transformé la conception des sites web, et de ce fait l'utilisation qu'en ont les internautes.

Au-delà de ces formulations marketing, nous allons tenter de voir plus précisément comment ces évolutions se manifestent aux internautes, et les changements topologiques qu'elles impliquent.

page 18

4.1 Des « applications Internet riches »...

L'une de ces évolutions porte sur l'interactivité des sites web. Ce ne sont plus seulement des pages statiques à l'image de celles d'un livre ou d'un magazine. En utilisant des technologies pré-existantes au web 2.0 comme le JavaScript et le Flash, les sites web ressemblent de plus en plus à des applications telles que celles que l'on trouve sur nos ordinateurs personnels : des sites web dynamiques répondant aux sollicitations de l'internaute.

page 22

De plus, la plupart des logiciels habituellement installés sur un ordinateur personnel sont transposés en version web, et deviennent accessibles depuis un navigateur web. On voit même apparaître des systèmes d'exploitation, comme Chrome OS, conçus entièrement selon ce principe. Ce mouvement, ce déplacement du logiciel installé sur l'ordinateur vers le web, est notamment une réponse aux soucis d'incompatibilité des logiciels, de licences et de mises à jour.

tome 1 § 1.4.1

En effet, plus besoin d'installation : une simple connexion à Internet et on dispose, *via* un navigateur web, de la plupart des applications traditionnelles : traitement de texte, tableur, messagerie électronique, agenda collaboratif, système de partage de fichiers, lecteur de musique, *etc.*

Ainsi *Google Drive* permet entre autres de rédiger des documents ou bien de faire sa comptabilité en ligne. Mais ce service permet également de la partager avec des amis, des collègues, *etc.*

1. L'exposé d'ouverture de la conférence de O'Reilly et Battelle sur le Web 2.0, cité par Wikipédia, 2014, *Web 2.0* [https://fr.wikipedia.org/wiki/Web_2.0] est un bel exemple de définition trop technique.

Certains vont même jusqu'à voir dans cette possibilité d'accéder à ces outils en ligne depuis « n'importe quel ordinateur, dans n'importe quel pays et à n'importe quelle heure » [2] une façon de concilier le travail avec d'éventuelles problémes médicaux, météorologique voir même en cas de pandémie… Plus besoin d'aller au bureau, « un ordinateur connecté à Internet suffit à reconstituer immédiatement l'environnement de travail. »

4.2 …et des clients devenus bénévoles

En arrivant sur le marché web, ces entreprises durent revoir leur modèle économique. L'audience de l'Internet grandissant, il n'était pas possible de financer un site web sur la seule publicité, tout en payant une armée de rédacteurs pour fournir du contenu en quantité toujours plus importante.

Les fournisseurs de services utilisèrent une technique déjà présente sur le web depuis longtemps : miser sur la participation des internautes. Ce sont dorénavant ceux-ci qui se chargent de rédiger le contenu qui alimente les sites. Les fournisseurs de services se contentent d'héberger les données et de fournir l'interface permettant d'y accéder, mais aussi et surtout d'ajouter de la publicité autour… et d'encaisser la monnaie.

Ainsi, la plateforme de partage de vidéo YouTube à, pendant de nombreuses années, permis à ses internautes de mettre en ligne et de visionner gratuitement les vidéos de leur choix sans contrepartie visible. Aujourd'hui, suite au succès et fort de son monopole, la plupart des personnes voulant visionner et partager des vidéos sont dépendantes de cette plateforme, ce qui permet alors à YouTube d'imposer petit à petit de la publicité. Au début, elle se situait sur un bandeau à côté de l'image, puis sur un bandeau transparent sur l'image et maintenant c'est tout simplement une vidéo incrustée au début de celle que l'on souhaite visionner [3].

Autre avantage de cette solution pour les fournisseurs de services, les internautes fournissent ainsi plus ou moins consciemment tout un ensemble de données [4] qu'il est ensuite possible de monnayer, notamment en constituant des profils de consommateurs et en adaptant les publicités affichées au public.

page 33

Il est ainsi courant que les internautes n'utilisent plus Internet uniquement pour télécharger des films ou aller y lire leur périodique favori. De plus en plus, par exemple *via* le remplissage de leur page Facebook, les internautes produisent du contenu et l'offrent pour ainsi dire aux hébergeurs ou autres entreprises qui fournissent ces services. L'internaute va « de lui-même » mettre en ligne la liste de la musique qu'il écoute, les photos de ses vacances au Mexique, ou encore ses cours d'histoire contemporaine pour les partager avec ses camarades de classe.

Bien sûr, en fournissant du contenu, on fournit aussi des informations sur soi, informations que les regards indiscrets des publicitaires et autres adversaires ne manqueront pas d'utiliser.

page 33
page 35

4.3 Centralisation des données

L'utilisation d'Internet comme espace de stockage de données va de pair avec la centralisation des données des internautes aux mains de quelques organisations, dans quelques lieux géographiques.

2. Lionel Damm et Jean-Luc Synave, 2009, *Entrepreneur 2.0, la boite à outils de la compétitivité… à petit frais* [http://www.confederationconstruction.be/Portals/28/UserFiles/Files/WP2guideentrepreneurweb20.pdf].

3. Jean-Baptiste Liouville, 2013, *YouTube : Changement des règles, oui, mais pour quelles raisons ?* [http://atelierdunumerique.com/youtube-changement-des-regles/].

4. Fanny Georges, Antoine Seilles, Jean Sallantin, 2010, *Des illusions de l'anonymat – Les stratégies de préservation des données personnelles à l'épreuve du Web 2.0*, Terminal numéro 105, Technologies et usages de l'anonymat à l'heure d'Internet.

L'utilisation d'applications en ligne signifie entre autres que les documents ne sont plus stockés sur un ordinateur personnel, un disque dur ou une clé USB. Ils se retrouvent sur des serveurs distants comme ceux de Google [5] ou d'Ubuntu One, dans des centres de traitement de données. Suffisamment loin de l'internaute, géographiquement comme techniquement, pour que l'on puisse douter du pouvoir qu'il a dessus. Une simple absence de connexion Internet et il devient impossible d'avoir accès à ses documents, à moins d'en avoir effectué une sauvegarde. Ce déplacement du stockage rend également impossible de pouvoir effacer avec certitude et de façon sécurisée les documents qui y sont placés.

tome 1 § 4.3

Cette tendance à faire migrer données et applications de l'ordinateur personnel vers Internet crée du même coup une « dépendance à la connexion ». Quand toute sa musique, son carnet d'adresses et les cartes de sa ville n'existent plus que par Internet, il devient plus difficile d'imaginer utiliser un ordinateur *hors connexion*. Or toute connexion à Internet ouvre des portes. Et plus un ordinateur est exposé, plus il est difficile de garantir sa sécurité – de l'anonymat de l'internaute qui l'utilise à la confidentialité des données qu'on lui confie.

page 33

Rien ne nous garantit non plus que nos données stockées en ligne soient bien gardées. Même si une organisation nous donne aujourd'hui tous les gages de sécurité (et encore, qu'est-ce qui nous le prouve ?) elle n'est de toute façon pas à l'abri, demain, de la découverte d'une faille, ou d'une erreur de configuration d'un programme qui donnerait accès à ces données à n'importe qui, comme ce fut le cas pour le service de stockage chiffré de données en ligne Dropbox [6].

page 44

Les entreprises à qui on confie nos données peuvent aussi supprimer notre compte [7], voire choisir de fermer leurs services sans que l'on y puisse rien - ou simplement faire faillite, ou se faire fermer par décision de justice comme dans le cas de Megaupload.

page 42

4.4 Mainmise sur les programmes

La plupart du temps, ces applications en ligne sont développées de manière plus fermée que les applications libres que l'on peut installer sur son ordinateur. Lorsque Google ou Facebook décident de modifier son interface ou de changer le fonctionnement du service, de « faire du rangement », l'internaute n'a pas son mot à dire.

tome 1 § 4.1

De plus, l'interactivité de ces applications web implique qu'une partie de leur programme soit exécuté sur l'ordinateur client (le nôtre), à travers des technologies comme JavaScript, Flash, ou encore Java. Ces technologies sont désormais activées, par défaut, dans nos navigateurs, et ceci pour tous les sites. C'est sympa, pratique, moderne. Mais ces technologies posent quelques problèmes quant à la sécurité de nos ordinateurs, et donc quant à la confidentialité de nos données… Il est cependant possible [8] de n'autoriser leur usage que site par site, en fonction de la confiance qu'on leur accorde.

page 26

tome 1 § 3.1

5. Le paragraphe *vos contenus et nos Services* des Conditions Générales d'Utilisation [https://www.google.com/intl/fr/policies/terms/] des services fournis par Google démontre assez clairement l'absence de pouvoir concret d'un utilisateur sur les contenus qu'il a stockés en ligne. « Ce qui est à vous, reste à vous » mais libre à Google d'en faire ce qu'il en a envie tant que vous laissez votre contenu sur ses serveurs.

6. Vincent Hermann, 2011, *Dropbox admet posséder un double des clés d'accès aux données* [http://www.pcinpact.com/breve/64460-dropbox-conditions-utilisation-chiffrement-securite.htm].

7. Owni, 2011, *Après 7 ans d'utilisation, il se fait supprimer son compte Google, donc les emails, le calendriers, les docs, etc.* [http://owni.fr/2011/08/29/google-suppression-compte-donnees-personnelles-vie-privee-god/].

8. Suivant le navigateur qu'on utilise, il existe des *extensions*, comme noscript [http://noscript.net], qui permettent de gérer ces paramètres.

4.5 De la centralisation à l'auto-hébergement décentralisé

Face à une centralisation toujours croissante des données et des applications, peut-on profiter des avantages d'un réseau participatif et interactif sans perdre le contrôle sur nos données ? Le défi paraît ardu. Mais des travaux sont en cours pour développer des applications web qui fonctionneraient de façon décentralisée chez chaque internaute au lieu d'être centralisées sur quelques serveurs. Des projets comme les médias sociaux de pair à pair, ownCloud [9], la FreedomBox [10], ou encore la distribution YunoHost [11] travaillent dans cette direction.

En attendant qu'ils soient aussi simples d'utilisation que les solutions proposées par les géants du web 2.0, il est d'ores et déjà possible, en mettant un peu les mains dans le cambouis, d'héberger soi-même la plupart des services qu'on souhaite offrir ou utiliser.

9. Owncloud [http://owncloud.org/] (en anglais)
10. Freedombox foundation [http://freedomboxfoundation.org] (en anglais).
11. Page francophone du projet YunoHost [https://doc.yunohost.org/#/index_fr]

Chapitre 5

Identités contextuelles

L'un des présupposés de ce *guide* est le désir que nos faits, gestes et pensées ne soient pas automatiquement, voire pas du tout, reliés à notre identité civile.

Pour autant, il peut être nécessaire ou simplement préférable de savoir à qui on s'adresse : pour entamer une discussion sur un forum ou envoyer des emails par exemple. Dans ces cas là, avoir une *identité*, c'est-à-dire être identifiable par notre correspondant, simplifie la communication.

5.1 Définitions

Pour commencer, deux définitions :

- l'*anonymat*, c'est ne pas laisser apparaître de nom ;
- le *pseudonymat*, c'est choisir et utiliser un nom différent de son nom civil.

De par son fonctionnement, il est très difficile d'être *anonyme* ou de rester un *pseudonyme* sur Internet.

5.1.1 Pseudos

Un *pseudo*, c'est une identité qui n'est pas celle assignée à un individu par l'état civil. On peut choisir de se faire appeler « Spartacus », « Amazone enragée », « Zigouigoui », ou même « Jeanne Dupont ». En conservant un même pseudonyme lors de différents échanges, nos interlocuteurs auront de bonnes chances de penser que les divers messages écrits par ce *pseudo* viennent de la même personne : ils pourront alors nous répondre, mais ne pourront pas venir nous casser la gueule en cas de désaccord.

Il faut néanmoins être conscient lors du choix d'un pseudonyme que celui-ci peut en lui-même être un indice qui permet de remonter à la personne qui l'utilise, au moins pour les personnes qui connaissent déjà ce pseudonyme par ailleurs.

5.1.2 Identité contextuelle

> *En reprenant le fil de notre histoire introductive, l'identité contextuelle correspondrait à « une ou plusieurs personnes publiant des informations sur le Maire du 10ème arrondissement », et la personne physique à Benoît.*

Que l'on discute avec des personnes avec qui on partage la passion de l'escalade, ou de notre projet professionnel avec un agent Pôle Emploi ou encore avec notre banquier, la teneur des propos, la manière dont on en parle n'est pas la même. D'un côté on sera plutôt exaltée, aventureuse, de l'autre plutôt sobre, sérieuse… on peut donc parler d'identité contextuelle.

Il en va de même lors de l'utilisation d'un ordinateur : quand on poste un message sur un forum de rencontre, quand on annonce une grosse soirée sur son compte Facebook ou quand on répond à un email de papa, on fait appel à différentes identités contextuelles. Celles-ci peuvent bien évidemment être mélangées et donc rejoindre une même identité composée des trois identités contextuelles mobilisées ci-dessus, la célibataire, la fêtarde, la fille de. Elles sont en définitive toutes constitutives des personnalités de leur propriétaire.

Une identité contextuelle est donc un fragment d'une « identité » globale censée correspondre à une personne physique, ou à un groupe. Tout comme une photographie est un instantané d'une personne ou d'un groupe, sous un certain angle, à un certain âge, *etc.*

page 25

Être absolument anonyme sur Internet, c'est très compliqué : comme on l'a vu, de nombreuses traces sont enregistrées *via* le réseau lors de son utilisation. Ce phénomène est d'autant plus vrai avec les médias sociaux pour lesquels la génération d'une identité unique et traçable est un fond de commerce [1]. Il est impossible de ne laisser aucune trace, mais il est peut-être possible de laisser des traces qui ne ramènent nulle part.

On rencontre des difficultés similaires lorsqu'on fait le choix du pseudonymat : plus on utilise un *pseudo*, plus les traces qu'on laisse s'accumulent. Des petits indices qui, une fois recoupés, peuvent permettre de révéler l'identité civile qui correspond à un pseudonyme.

5.2 De l'identité contextuelle à l'identité civile

Il existe différentes manières, plus ou moins offensives, de mettre à mal un pseudonyme ou de révéler le lien entre une identité contextuelle et la ou les personnes physiques qui l'utilisent.

5.2.1 Le recoupement

page préc.

En partant de l'exemple des trois identités contextuelles, il est légitime de se demander ce que jongler entre ces différentes identités implique en termes d'anonymat. En imaginant qu'on utilise un pseudonyme et non son état civil, il peut être plus pertinent d'avoir une identité, donc un *pseudo*, dans chaque contexte : une pour les sites de rencontres, une autre pour les médias sociaux, et une pour les relations familiales, *etc.* afin d'éviter les recoupements. Si les informations émanant des dites identités ne sont pas compartimentées, c'est-à-dire si un même pseudo est utilisé, leur recoupement permet de réduire le nombre de personnes à qui elles peuvent correspondre. Il devient alors plus facile de faire le lien entre une présence numérique et une personne physique, et donc de mettre un nom sur l'identité contextuelle correspondante.

Considérons par exemple un internaute qui utilise le pseudonyme *bruce76* sur un blog où il dit être végétarien et aimer les films d'action. Il n'existe qu'un certain nombre de personnes correspondant à ces critères. Ajoutons à cela le fait que ce même pseudonyme est utilisé pour organiser une fiesta dans telle ville *via* un réseau social et pour communiquer par email avec Mme Unetelle. Il n'y a sans doute pas beaucoup de personnes végétariennes, aimant les films d'actions, organisant une fête dans cette même ville et communiquant par email avec Mme Unetelle.

Plus les utilisations d'un pseudonyme sont nombreuses et variées, plus le nombre de personnes pouvant correspondre à ce pseudonyme est restreint. Il est donc possible, en recoupant les utilisations d'un même pseudonyme par exemple, d'affaiblir voire de casser le pseudonymat.

C'est ce que nombre d'utilisateurs d'AOL découvrirent à leurs dépends lors de la publication de plus de trois mois de résultats d'utilisation du moteur de recherche de la

1. ippolita, 2012, *J'aime pas Facebook*, Payot [http://www.ippolita.net/fr/jaime-pas-facebook]

firme [2]. Nombre de chercheurs purent facilement briser le faible pseudonymat appliqué par AOL sur ces données. Le gouverneur de l'état du Massachusetts a lui aussi fait les frais de ces recoupements lorsque son dossier médical, soit-disant anonymisé, a pu être identifié parmi ceux de tous les citoyens de cet état. La chercheuse ayant effectuée cette démonstration de désanonymisation de données poussa l'ironie jusqu'à lui envoyer son dossier médical par la poste [3].

5.2.2 Corrélation temporelle

Procédé un peu plus technique cette fois-ci, la corrélation temporelle permet également de briser ou d'affaiblir un peu plus l'anonymat ou le pseudonymat. En effet, si dans un intervalle de temps réduit, il y a connexion vers la boîte mail `amazone@exemple.org` ainsi que `jeanne.dupont@courriel.fr`, la probabilité que ces deux adresses emails soient aux mains de la même personne augmente, et ce d'autant plus si cette observation se répète. Diverses parades, répondant à des besoins divers, seront explicitées plus loin.

5.2.3 Stylométrie

Il est possible d'appliquer des analyses statistiques sur la forme de n'importe quel type de données, et notamment aux textes. En analysant [4] différentes caractéristiques d'un texte, comme la fréquence des mots-outils [5], la longueur des mots, des phrases et des paragraphes, la fréquence des signes de ponctuation, on peut corréler des textes anonymes avec d'autres textes, et en retirer des indices sur leur auteur.

Ce type d'analyse fut par exemple utilisé lors du procès de Theodore Kaczynski [6] pour accréditer le fait qu'il soit l'auteur du manifeste « La société industrielle et son avenir » [7].

Les auteurs d'une étude récente [8] ont cherché à « simuler une tentative d'identification de l'auteur d'un blog publié de manière anonyme. Si l'auteur est suffisamment prudent pour éviter de révéler son adresse IP ou tout autre identifiant explicite, son adversaire (par exemple un censeur gouvernemental) peut se pencher sur l'analyse de son style d'écriture ». Leurs conclusions montrent que la stylométrie permet de réduire fortement, parmi de très nombreuses possibilités, le nombre d'auteurs possibles d'un texte anonyme – la précision augmentant évidemment avec le nombre d'échantillons « signés », c'est-à-dire dont l'auteur est connu, fournis au logiciel d'analyse.

Le plus souvent, cela leur permet de réduire la taille de l'ensemble des auteurs possibles de 100 à 200 sur 100 000 initialement. « [...] ajouté à une autre source d'information, cela peut être suffisant pour faire la différence entre l'anonymat et l'identification d'un auteur ». Dans 20 % des cas, il est même possible d'identifier directement l'auteur anonyme.

La particularité de ce travail est qu'il dépasse le cadre de petits échantillons (une centaine de possibilités) auxquels s'étaient cantonnées les études précédentes, pour

2. Nate Anderson, 2006, *AOL releases search data on 500,000 users* [http://arstechnica.com/uncategorized/2006/08/7433/] (en anglais).
3. Paul Ohn, 2009, *Broken Promises of Privacy : Responding to the Surprising Failure of Anonymization* [http://www.uclalawreview.org/pdf/57-6-3.pdf] (en anglais).
4. Par exemple grâce à des logiciels comme *The Signature Stylometric System* [http://www.philocomp.net/?pageref=humanities&page=signature] ou *Java Graphical Authorship Attribution Program* [http://www.jgaap.com/] (liens en anglais).
5. Les mot-outils sont des mots dont le rôle syntaxique est plus important que le sens. Il s'agit typiquement de mots de liaison [https://fr.wikipedia.org/wiki/Mot-outil]
6. Kathy Bailey, 2008, *Forensic Linguistics in Criminal Cases*, Language in Social Contexts [http://ksbailey.writersresidence.com/samples/forensic-linguistics-in-criminal-cases] (en anglais).
7. Theodore Kaczynski, 1998, *La société industrielle et son avenir* [http://www.fichier-pdf.fr/2012/12/20/kaczynski/kaczynski.pdf]
8. Hristo Paskov, Neil Gong, John Bethencourt, Emil Stefanov, Richard Shin, Dawn Song, 2012, *On the Feasibility of Internet-Scale Author Identification* [http://randomwalker.info/publications/author-identification-draft.pdf] (en anglais).

s'intéresser à l'identification de l'auteur parmi un très grand nombre de possibilités ; en d'autres termes, il démontre que la stylométrie peut être employée pour confirmer l'origine d'un texte sur la base d'un très grand nombre d'échantillons.

Cependant, écrire en essayant de masquer son style, sans expertise particulière, semble permettre de rendre inefficaces les analyses stylométriques. Imiter le style de quelqu'un d'autre permet même de les tromper dans plus de la moitié des cas [9].

D'autres chercheurs développent des logiciels qui suggèrent les modifications à effectuer pour anonymiser un texte [10].

5.3 La compartimentation

Comme on vient de le voir, de nombreuses possibilités d'attaques permettent de faire correspondre une identité civile et une identité contextuelle. L'utilisation d'un seul et même nom pour ses différentes activités est sans doute la pratique la plus à même de nous confondre.

Face à cela, il est donc important de bien réfléchir à l'usage que l'on a de ses pseudonymes. Il est souvent dangereux de mélanger plusieurs identités contextuelles sous un même pseudo. La meilleure prévention reste de les séparer clairement dès le départ afin de limiter les ennuis par la suite. Après tout, une pratique ou une identité qui peut être utilisée à un moment donné peut d'un coup se transformer en source de problèmes en raison de conditions extérieures qu'il n'est pas forcément possible d'anticiper ou de maîtriser.

page 54

Cependant, ces pratiques ne sont pas toujours faciles à mettre en place. Car en plus des techniques décrites précédemment, la séparation entre ces différentes identités contextuelles dépend de beaucoup d'autres paramètres. Notamment des relations que l'on établit avec d'autres personnes, que ces relations soient numériques ou non. Il n'est pas forcément facile d'avoir une identité contextuelle différente pour absolument chacune des facettes de sa personnalité ou chacune de ses activités, ni d'éviter que certaines d'entre elles ne se recoupent. Ces identités évoluent au gré des activités qu'on leur attribue et au fil du temps. Plus longtemps on les utilise, plus leur séparation a tendance à s'amenuiser. Il est donc souvent difficile d'équilibrer et de mesurer les efforts nécessaires à la mise en place des multiples identités contextuelles avec les bénéfices escomptés. D'autant plus qu'il est généralement compliqué de faire marche arrière dans ce domaine.

Certains outils tels les médias sociaux les rendent même quasiment impraticables en imposant une transparence absolue.

5.4 Les médias sociaux : centralisation de fonctions et identité unique

Les médias sociaux tendent en effet à centraliser des fonctions qui étaient auparavant assurées par différents outils, de l'échange de messages à la publication de nouvelles, en passant par les groupes de discussion. Ils tendent à se substituer à la fois à l'email, à la messagerie instantanée, aux blogs ainsi qu'aux forums.

Dans le même temps se développent de nouvelles fonctions, comme une certaine vie relationnelle numérique où l'existence d'une communication prime sur son contenu,

9. M. Brennan, R. Greenstadt, 2009, *Practical attacks against authorship recognition techniques*, dans *Proceedings of the Twenty-First Innovative Applications of Artificial Intelligence Conference* [http://www.cs.drexel.edu/~greenie/brennan_paper.pdf] (en anglais).

10. Andrew W.E. McDonald, Sadia Afroz, Aylin Caliskan, Ariel Stolerman, Rachel Greenstadt, 2012, *Use Fewer Instances of the Letter "i" : Toward Writing Style Anonymization*, The 12th Privacy Enhancing Technologies Symposium [https://www.cs.drexel.edu/~sa499/papers/anonymouth.pdf] (en anglais).

poussée à son paroxysme avec les « pokes », ces messages sans contenu [11]. Le web 2.0 encourage l'expression sur des sujets qui étaient auparavant considérés comme intimes [12].

Finalement, pas grand-chose de bien nouveau, si ce n'est la centralisation de nombreuses fonctions et de pratiques variées vers un outil unique. C'est d'ailleurs le côté « tout-en-un » de ces plateformes, le graphisme ainsi que la facilité d'usage qui en font le succès. Mais cette centralisation pose question quant aux conséquences de l'utilisation de ces outils sur nos intimités.

La pression sociale pour utiliser les médias sociaux est très forte dans certains milieux : lorsque des groupes les utilisent pour la majorité de leurs communications, des messages interpersonnels aux invitations en passant par la publication d'informations, ne pas participer aux médias sociaux, c'est être marginalisé. Le succès de ces sites repose sur « l'effet de réseau » : plus il y a de personnes qui les utilisent, plus il est important d'y être présent.

Mais dans le même temps, ces médias sociaux permettent aussi de s'évader de ces pressions de groupes et d'assumer ou d'expérimenter plus facilement certaines parts de sa personnalité qui ne sont pas forcément tolérées par ces groupes.

La centralisation de toutes les activités sur une seule plateforme rend extrêmement difficile l'usage de pseudonymes différents pour différentes identités contextuelles. En effet, en mettant toutes les informations au même endroit, le risque de recoupement de différentes identités contextuelles est maximisé. Nombre de médias sociaux demandent une identité unique, celle correspondant à l'état civil d'une personne physique. C'est là une différence clé par rapport à un modèle où un individu peut avoir plusieurs blogs avec des tons et des contenus différents, chacun sous un pseudonyme différent. De plus, à l'instar des sites de rencontres, où plus on est honnête, meilleurs sont les résultats, ici plus on fournit du contenu, plus on utilise cette plateforme, meilleures sont les interactions.

Ceci est d'autant plus vrai qu'utiliser son nom d'état civil fait partie des règles de réseaux comme Facebook, qui met en place différents mécanismes pour traquer les pseudonymes [13]. Ces entreprises poussent jusqu'au bout le *business model* de la publicité ciblée et de la vente de profils : ils « mettent en place différents procédés techniques de captation de l'identité des usagers, depuis l'identité fondée sur leurs déclarations, jusqu'à l'identité agissante [14] et l'identité calculée fondée sur l'analyse de leurs comportements : sites visités, nombre de messages, *etc.* Il apparaît que l'anonymat total devient impossible dans un univers virtuel où les usagers sont avant tout des consommateurs qu'il s'agit d'observer. » [15]

Ainsi, en juillet 2011, Max Schrems a réussi à obtenir l'ensemble des données dont Facebook dispose sur lui en invoquant une directive européenne. Le dossier qu'il a

11. Fanny Georges, 2008, *Les composantes de l'identité dans le web 2.0, une étude sémiotique et statistique*, Communication au 76ème congrès de l'ACFAS : Web participatif : mutation de la communication ?, Québec, Canada [http://hal.archives-ouvertes.fr/hal-00332770/]

12. Alain Rallet et Fabrice Rochelandet, 2010, *Exposition de soi et décloisonnement des espaces privés : les frontières de la vie privée à l'heure du Web relationnel*, Terminal numéro 105, Technologies et usages de l'anonymat à l'heure d'Internet [http://ead.univ-angers.fr/~granem08/IMG/pdf/Rochelandet.pdf]

13. Nikopik, 2012, *Facebook et la délation* [http://www.nikopik.com/2012/07/facebook-vous-demande-de-denoncer-vos-amis-a-pseudonyme.html].

14. Identité agissante : « messages notifiés par le Système concernant les activités de l'utilisateur ». « Par exemple, " a modifié sa photo de profil ", " est désormais ami avec " » dans l'historique de Facebook ou Linkedin » (Fanny Georges, Antoine Seilles, Jean Sallantin, 2010, *Des illusions de l'anonymat – Les stratégies de préservation des données personnelles à l'épreuve du Web 2.0*, Terminal numéro 105, Technologies et usages de l'anonymat à l'heure d'Internet)

15. Chantal Enguehard, Robert Panico, 2010, *Approches sociologiques*, Terminal numéro 105, Technologies et usages de l'anonymat à l'heure d'Internet [http://www.revue-terminal.org/www/articles/105/introPartie2Anonymat105.pdf]

reçu comprend 1222 pages[16], qui incluent non seulement l'ensemble des informations disponibles sur son profil, mais aussi tous les évènement auxquels il a été invité (y compris les invitations déclinées), tous les messages envoyés ou reçus (y compris les messages supprimés), toutes les photos chargées sur Facebook accompagnées de méta-données concernant notamment la géolocalisation, tous les « pokes » émis ou reçus, tous les « amis » (y compris les « amis » effacés), les journaux de connexions à Facebook (incluant l'adresse IP et sa géolocalisation), toutes les « machines » (identifiées par un cookie) utilisées par un profil, ainsi que les autres profils utilisant les mêmes « machines » ou encore la localisation de sa dernière connexion connue à Facebook (longitude, latitude, altitude).

tome 1 § 2.6

page 26

Enfin, malgré les déclarations du fondateur de Facebook, comme quoi l'ère de la vie privée est révolue[17], nombre de stratégies restent à développer, à remanier, afin de jouer avec les différentes marges encore d'actualité. Et ceci dans l'optique d'avoir un peu de prise sur ces questions fondamentales : « Qu'est-ce que l'on souhaite montrer ? », « Qu'est-ce que l'on accepte de rendre visible ? » et « Qu'est-ce que l'on veut cacher et à quel prix ? ».

16. Europe versus Facebook, 2012, *Facebook's Data Pool* [http://europe-v-facebook.org/EN/Data_Pool/data_pool.html] (en anglais).
17. Bobbie Johnson, 2010, *Privacy no longer a social norm, says Facebook founder* [http://www.theguardian.com/technology/2010/jan/11/facebook-privacy] (en anglais).

Chapitre 6

Cacher le contenu des communications : la cryptographie asymétrique

Dans le premier tome de ce guide, nous avons vu que la piste la plus sérieuse pour protéger des données des regards indiscrets est le chiffrement : il permet de les rendre illisibles pour toute personne qui n'a pas la *clé secrète*.

tome 1 ch. 5

6.1 Limites du chiffrement symétrique

Dans le cadre du chiffrement symétrique, c'est une même clé secrète qui permet à la fois d'effectuer le chiffrement et le déchiffrement.

Le chiffrement symétrique est tout à fait adapté pour chiffrer une clé USB ou un autre support de stockage.

tome 1 § 5.1.4

Cependant, dans le cas d'une communication, lorsque la personne qui devra déchiffrer les données n'est pas la même que celle qui les a chiffrées, plusieurs problèmes se posent :

Tout d'abord, il faut une nouvelle clé secrète pour chaque couple émetteur/récepteur : si je veux pouvoir échanger des messages chiffrés avec ma sœur d'une part, et un ami d'autre part, j'aurai besoin d'utiliser deux clés différentes, sans quoi ma sœur pourrait déchiffrer les messages que j'échange avec mon ami, et inversement. Du moins dans le cas où l'un d'eux *tombe* sur les messages que j'échange avec l'autre.

De plus, expéditeur et destinataire doivent se mettre d'accord sur une clé secrète et se l'échanger de façon confidentielle. Si un adversaire entrait en possession de la clé secrète, il pourrait déchiffrer tous nos échanges passés, mais aussi futurs. Il serait en effet nécessaire mais très difficile que notre interlocuteur soit prévenu de façon sûre (en nous authentifiant) que le secret a été éventé. Supposons que l'on reçoive un message disant « le secret n'est plus sûr » : si c'était vrai, il faudrait arrêter d'utiliser ce moyen de communication. De plus, un adversaire désirant perturber nos communications pourrait également délivrer ce message et ainsi parvenir à ses fins sans beaucoup d'efforts.

La crytographie asymétrique répond à ces limites… mais en présente d'autres.

6.2 Une solution : la cryptographie asymétrique

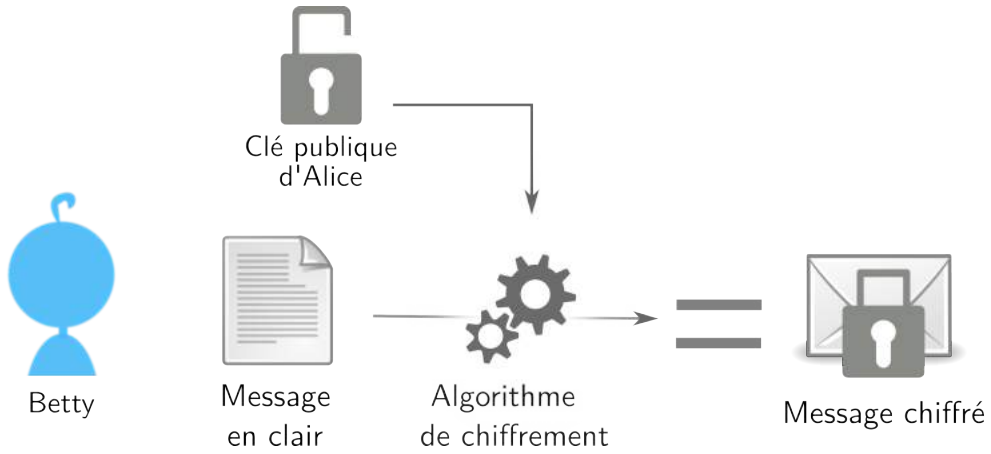
Dans les années 70, des mathématiciens ont révolutionné la cryptographie en trouvant une solution aux problèmes posés par le chiffrement symétrique, en créant le chiffrement *a*symétrique. « Asymétrique » car il utilise, pour déchiffrer un message, une clé différente de celle qui a permis de le chiffrer.

tome 1 § 5.1

Prenons l'exemple d'Alice, qui souhaite recevoir un message chiffré de la part de Betty. Elle envoie à Betty un cadenas ouvert, dont elle garde précieusement la clé :

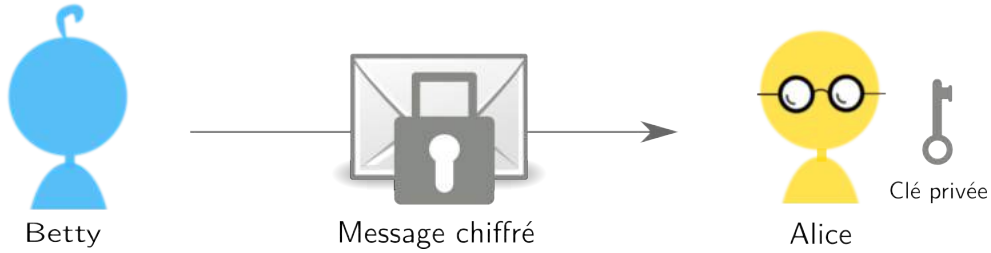
Alice envoie sa clé publique

Betty place alors son message dans une boîte, et utilise le cadenas pour fermer la boîte – elle n'a pas besoin de la clé du cadenas pour cela :



Betty chiffre un message

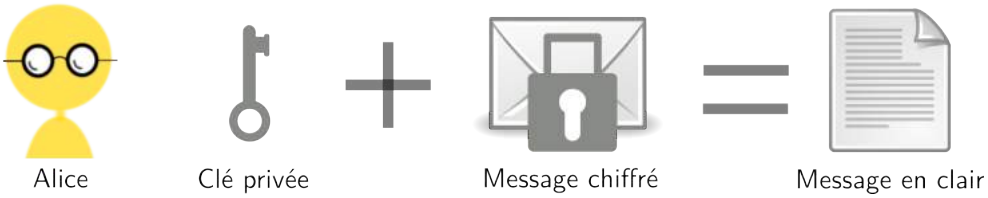
Betty renvoie alors la boîte contenant le message, protégée par le cadenas fermé, à Alice :



Betty envoie un message chiffré

Grâce à la clé, qu'elle a toujours gardée sur elle, Alice peut ouvrir le cadenas :

On le voit, grâce à la cryprographie asymétrique, la seule chose qui circule sur le réseau est un cadenas ouvert, puis un cadenas fermé. Et si une personne mal intentionnée tombe sur le cadenas ouvert, ce n'est pas très grave : cela ne lui permet pas d'ouvrir un cadenas fermé.



Alice déchiffre un message chiffré

6.2.1 Clé publique, clé privée

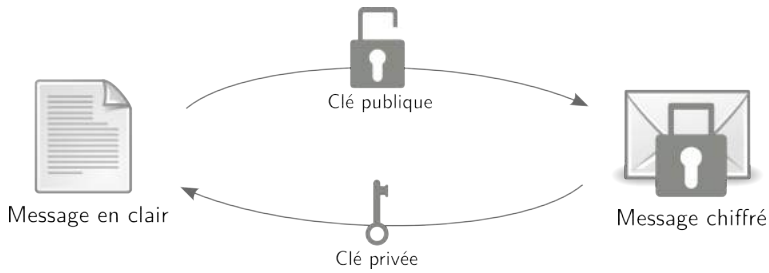
Ce type de chiffrement est aussi appelé « chiffrement à clé publique ». Le cadenas ouvert d'Alice est sa *clé publique*, ainsi appelée car il n'est pas nécessaire de la cacher : elle peut être rendue publique, par exemple publiée sur Internet, afin que toute personne qui souhaite écrire à Alice de manière chiffrée puisse se la procurer.

Au contraire, la clé du cadenas d'Alice, qui sert à ouvrir les cadenas fermés protégeant les messages, ne doit jamais tomber dans les mains de l'adversaire. Alice la garde donc précieusement, à l'abri des regards indiscrets : on l'appelle la *clé privée*.

Clé publique et clé privée forment la *paire de clés* d'Alice. Chaque entité qui souhaite pouvoir se faire envoyer des messages chiffrés asymétriquement doit disposer de sa propre paire de clés.

6.2.2 Une affaire de nombres premiers...

Dans la réalité, la clé publique et la clé privée sont des nombres. Ce qu'une clé permet de chiffrer, l'autre permet de le déchiffrer :



Mais comment est-il possible que la clé publique permette de chiffrer un message sans permettre de le déchiffrer ? La cryptographie asymétrique repose en fait sur des problèmes mathématiques extrêmement difficiles à résoudre. L'algorithme de chiffrement RSA, par exemple, repose sur la « factorisation de nombres entiers ».

Étant donné le nombre 12, il est simple de le décomposer en $2 \times 2 \times 3$. De même, 111 est égal à 3×37. En revanche, comment décomposer le nombre suivant, composé de 232 chiffres ?

1230186684530117755130494958384962720772853569595334792197322452151726400
5072636575187452021997864693899564749427740638459251925573263034537315482
6850791702612214291346167042921431160222124047927473779408066535141959745
9856902143413

Le résultat est le produit de deux nombres premiers composés chacun de 116 chiffres.

Ce problème de factorisation d'entiers est étudié depuis plus de 2000 ans par des mathématiciens ; pourtant, aucune solution pratique n'a encore été trouvée : la meilleure solution connue est d'essayer avec tous les nombres premiers possibles.

Avec un ordinateur actuel, ce calcul serait beaucoup plus long que la durée d'une vie humaine[1]. Les nombres les plus difficiles à factoriser sont les produits de deux grands nombres premiers. On choisira donc des nombres suffisamment grands pour que même avec des ordinateurs extrêmement puissants, la factorisation ne puisse pas se faire en un temps réaliste.

Faire confiance à cette méthode revient donc à faire le pari que son adversaire dispose d'une puissance de calcul relativement limitée. La taille des clés, qui se mesure en bits, est d'une importance capitale. En effet, si on considère qu'une clé asymétrique de 2048 bits[2] est sûre jusqu'en 2020[3], une clé de 512 bits se casse en quelques mois avec un ordinateur personnel haut de gamme actuel[4]. Il faut garder à l'esprit que ce qui est « cassable » par un ordinateur en 10 ans pourrait l'être en 1 an avec 10 ordinateurs identiques au premier.

De plus, si un jour une personne résout ce problème mathématique, il sera possible de déchiffrer sans trop de difficulté les échanges chiffrés qui auront ont été enregistrés – ce type de collecte et de stockage fait partie entre autres des activités de la NSA, agence de renseignement états-unienne[5]. Beaucoup de secrets militaires et commerciaux seraient alors révélés à ceux qui auront accès à ces enregistrements. En d'autres termes, on peut imaginer une sacrée pagaille entre entreprises concurrentes et agences de renseignements ennemies…

En attendant, les attaques utilisées à l'heure actuelle sur les systèmes de cryptographie asymétrique ciblent la façon de le mettre en œuvre dans tel ou tel logiciel, ou une erreur dans son code source, et non le principe mathématique du système.

tome 1 § 4.1.1

6.3 Signature numérique

Les paires de clés utilisées pour la cryptographie asymétrique peuvent aussi être utilisées pour prouver l'authenticité d'un message. Comment cela fonctionne-t-il ? Reprenons l'exemple de Betty envoyant un message à Alice. Cette fois, Betty veut signer numériquement son message afin qu'Alice puisse être sûre qu'elle en est bien l'auteur.

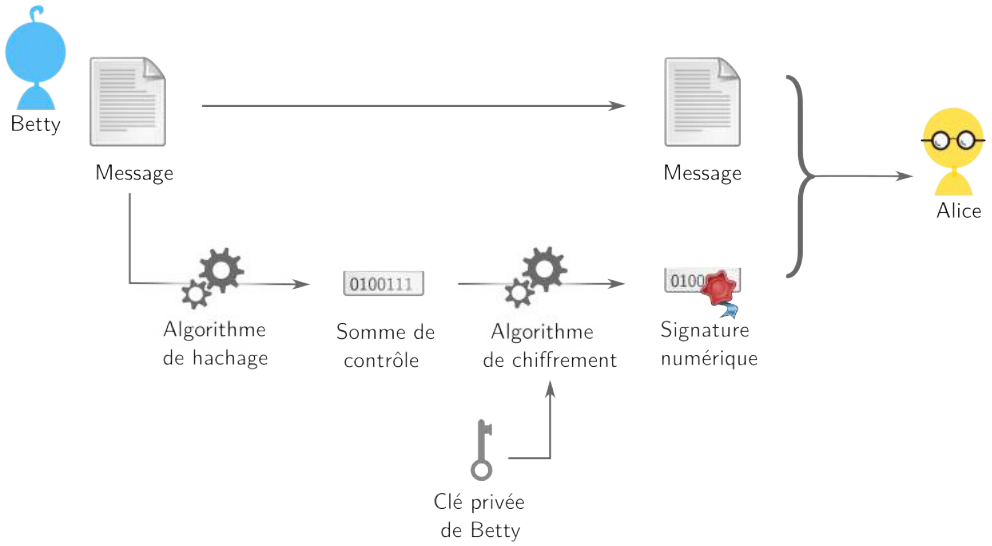
tome 1 § 5.2

Dans le premier tome de ce guide, on a parlé des sommes de contrôle, ou empreintes : un nombre qui permet de vérifier l'intégrité d'un message. Cette empreinte va également servir à signer des données numériques. Dans un premier temps, l'ordinateur de Betty calcule une *empreinte* du message qu'elle va envoyer à Alice.

Ensuite, cette empreinte est chiffrée avec la clé privée de Betty : c'est la *signature numérique*. Eh oui : l'empreinte est chiffrée avec la clé privée de Betty, dont elle est la seule à disposer, et non avec la clé publique d'Alice. Cette signature sert en effet à authentifier l'expéditeur, et non le destinataire. Or on vient de voir que clé publique et clé privée étaient en fait deux nombres choisis de telle façon que l'un permette de déchiffrer ce que l'autre a chiffré. Rien n'empêche donc de chiffrer quelque chose avec la clé privée. C'est alors la clé publique qui va permettre de le déchiffrer.

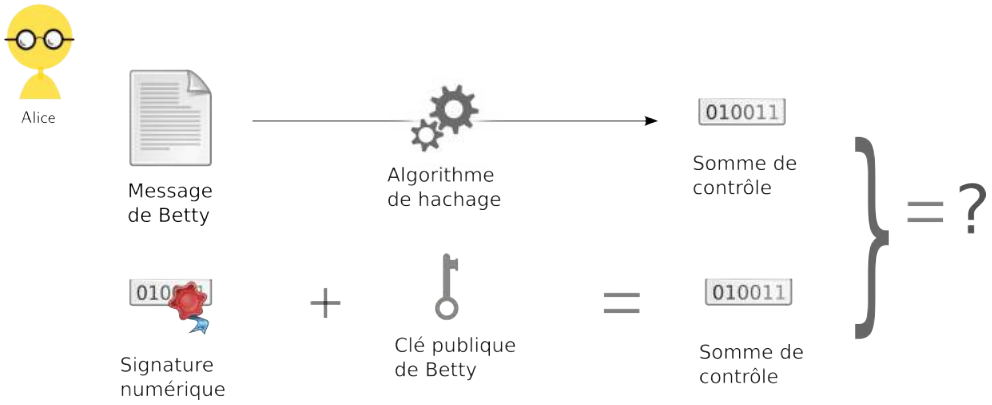
Betty envoie alors le message accompagné de sa signature à Alice.

1. La factorisation de ce nombre de 768 bits en 2010 a nécessité 20^10 opérations. Les chercheurs qui l'ont réalisée estiment que le calcul aurait pris environ 2000 ans sur un AMD Opteron à 2.2 GHz, ce qui correspond à plusieurs centaines d'années sur un processeur dernier cri (Thorsten et al., 2010, *Factorization of a 768-bit RSA modulus* [http://eprint.iacr.org/2010/006.pdf] – en anglais).
2. Un bit est un chiffre binaire (0 ou 1). Pour en savoir plus, voir Wikipédia, 2014, *Bit* [https://fr.wikipedia.org/wiki/Bit]
3. Agence nationale de la sécurité des systèmes d'information, 2010, *Mécanismes cryptographiques Règles et recommandations concernant le choix et le dimensionnement des mécanismes cryptographiques* [http://www.ssi.gouv.fr/IMG/pdf/RGS_B_1.pdf]
4. S. A. Danilov, I. A. Popovyan, 2010, *Factorization of RSA-180* [http://eprint.iacr.org/2010/270.pdf] (en anglais).
5. Nicole Perlroth, Jeff Larson et Scott Shane, 2013, *N.S.A. Able to Foil Basic Safeguards of Privacy on Web*, The New York Times [http://www.nytimes.com/2013/09/06/us/nsa-foils-much-internet-encryption.html] (en anglais).



Betty signe un message

Pour vérifier la signature, l'ordinateur d'Alice va lui aussi calculer l'empreinte du message et déchiffrer en parallèle la signature.



Alice vérifie le message

Puisqu'elle est chiffrée avec la clé privée de Betty, la clé publique de Betty suffit pour déchiffrer cette signature. Si l'empreinte du message reçu correspond à la signature déchiffrée (celle-ci n'étant rien d'autre, comme on l'a dit, que l'empreinte du message calculée par l'ordinateur de Betty), Alice est sûre de l'authenticité du message qu'elle a reçu. En effet, Betty garde sa clé privée en lieu sûr. Elle est donc la seule à avoir pu chiffrer l'empreinte qu'Alice a déchiffré avec la clé publique de Betty.

Le contrepoint négatif de cette certitude est que le possesseur d'une clé privée pourra plus difficilement nier être l'auteur du message.

6.4 Vérifier l'authenticité de la clé publique

La cryptograhie asymétrique permet ainsi de chiffrer et de signer des messages sans avoir besoin de s'échanger préalablement un secret partagé.

Cependant, elle ne résout pas une question importante : comment s'assurer que je possède bien la *véritable* clé publique de mon destinataire, et que ce n'est pas un

usurpateur qui m'a fourni une fausse clé publique pour pouvoir intercepter mes messages, tout en me donnant une fausse impression de sécurité ?

6.4.1 L'attaque de l'homme du milieu

Reprenons l'exemple d'Alice qui souhaite recevoir un message chiffré de la part de Betty, en présence d'une adversaire Carole qui peut avoir accès aux messages échangés :

- Alice commence par envoyer sa clé publique à Betty. Carole peut la lire.
- Betty chiffre son message avec la clé publique qu'elle a reçue, puis l'envoie à Alice.
- Carole qui ne possède pas la clé privée d'Alice, mais seulement sa clé publique, ne peut pas déchiffrer le message.
- Alice elle, peut déchiffrer le message à l'aide de la clé privée qu'elle garde précieusement.

Cependant si Carole est en mesure de modifier les échanges entre Alice et Betty, les choses se corsent :

- Lorsqu'Alice envoie sa clé publique à Betty, Carole l'intercepte et renvoie à Betty, en lieu et place de celle d'Alice, une clé publique dont elle détient la clé privée correspondante.
- Betty chiffre son message avec la clé publique qu'elle a reçue, puis l'envoie à Alice. Mais la clé qu'elle a reçue appartenait à Carole : elle l'a substituée à celle d'Alice.
- Carole intercepte à nouveau le message. Mais cette fois, il est chiffré avec sa clé publique, dont elle a la clé privée. Elle peut donc déchiffrer le message pour le lire et éventuellement le modifier. Puis elle chiffre à nouveau le message avec la véritable clé publique d'Alice, avant de l'envoyer à Alice.
- Alice peut alors déchiffrer le message avec sa clé privée, sans se rendre compte de rien.

Ainsi, Betty est persuadée d'utiliser la clé d'Alice, alors qu'elle utilise en réalité celle de Carole. De la même manière, Carole peut usurper la clé publique de Betty et falsifier la signature du message transmis par Betty à Alice. Alice recevra un message chiffré et dûment signé… par Carole.

On appelle cette attaque *l'attaque de l'homme du milieu* (*Man in the Middle attack*, ou *MitM*, en anglais)[6]. Dans notre exemple, Carole était l'« homme du milieu », capable de lire et de modifier la communication chiffrée en se faisant passer, aux yeux de chaque partie de la communication, pour l'autre.

Un adversaire peut se positionner en *homme du milieu* par différent biais.

Le fournisseur d'accès à Internet est par exemple particulièrement bien placé, car tout le trafic passera obligatoirement par lui. De même un *gros* nœud du réseau par lequel passe une quantité importante du trafic sera en bonne mesure de mettre en place cette attaque[7]. Enfin un adversaire ayant accès au réseau local que vous utilisez pourra toujours faire transiter le trafic réseau par son ordinateur utilisant pour cela des techniques plus spécifiques[8].

Pour se prémunir contre cette attaque, il faut que Betty ait une façon de vérifier que la clé publique qu'elle utilise est bien celle d'Alice. Si la clé publique n'est pas une information confidentielle, il faut donc toutefois s'assurer de son *authenticité* avant de l'utiliser.

Parfois, la façon la plus simple, pour Betty, est de rencontrer Alice afin de vérifier que la clé publique dont elle dispose est bien la sienne. Peu importe que Carole

6. [Wikipédia, 2014, *Attaque de l'homme du milieu*](https ://fr.wikipedia.org/wiki/Attaque_de_l'homme_du_milieu)
7. reflets.info, 2011, *#OpSyria : Bluecoat au coeur d'attaque MITM de grande envergure ?* [http://reflets.info/opsyria-bluecoat-au-coeur-dattaque-mitm-de-grand-envergure/]
8. Wikipédia, 2014, *ARP poisoning* [https://fr.wikipedia.org/wiki/ARP_poisoning]

soit présente au moment de cette rencontre : seule une vérification de clé *publique* aura lieu, et aucun secret ne va être échangé (à part que Betty et Alice souhaitent communiquer, mais ça, vu sa position, Carole peut le savoir d'autres façons). Une fois cette vérification faite, du chiffrement de *bout-à-bout* pourra être mis en place entre Alice et Betty. Le chiffrement est dit de *bout-à-bout* lorsqu'il a lieu entre la source et la destinataire d'une communication électronique, et cela sans interruption. Le chiffrement a lieu dans l'ordinateur d'Alice et le déchiffrement dans celui de Betty. Entre les deux, un message dont le contenu sera chiffré circulera ; seul l'en-tête de la communication, que ce soit une requête HTTP ou un email, circulera *en clair*.

page 29

Cependant, il arrive souvent que Betty ne puisse pas rencontrer Alice – a fortiori si elle ne la connaît pas : si elle rencontre une personne qui se présente comme étant Alice, Betty ne peut pas être sûre qu'il s'agit bien d'Alice. Or, c'est généralement le cas lorsqu'on veut chiffrer ses connexions vers un site web : on ne connaît pas les personnes qui sont derrière.

6.4.2 Infrastructure à clé publique

La première solution couramment utilisée est de disposer d'autorités de confiance qui certifient les clés publiques en les signant numériquement : on parle de *certificats*. Alice demande à l'autorité de certifier sa clé publique, souvent moyennant finances. L'autorité vérifie l'identité d'Alice, par exemple en lui demandant sa carte d'identité, puis signe numériquement sa clé. Avant d'utiliser la clé d'Alice, Betty (ou son ordinateur) vérifie qu'elle est bien signée par une autorité qu'elle considère comme digne de confiance. On parle d'infrastructure à clé publique (*public key infrastructure*, ou *PKI* en anglais).

page 62
page 61

C'est le principe qui est couramment utilisé pour authentifier les sites web ou les serveurs d'email avec lesquels l'ordinateur établit une connexion chiffrée. Les enjeux les plus courants lors de l'établissement d'une connexion chiffrée vers un site web sont la protection de mots de passe – pour se connecter à son compte email par exemple – ou la protection de données bancaires – pour effectuer des achats sur des sites de vente en ligne. Le protocole utilisé pour ce type de chiffrement est appelé TLS (anciennement SSL)[9].

page 18
page 11

Cependant, une telle solution ne fait que déplacer le problème : il faut faire confiance à l'autorité de certification. En général, ce sont des entreprises commerciales, et plus rarement des administrations.

Ainsi Microsoft, Apple et Mozilla incluent chacun des autorités de certification de gouvernements parmi les autorités de certification reconnues par leurs navigateurs web[10]. Mozilla Firefox inclut notamment des autorités de certifications de gouvernements (chinois, français, néerlandais, catalan, japonais), d'entreprises de certification (Verisign, GoDaddy), mais aussi d'entreprises de télécommunications (Deutsche Telecom, Hongkong Post)[11].

Ces gouvernements, qui peuvent souvent se positionner en *homme du milieu*, ont le pouvoir de désigner n'importe quel certificat comme valide pour un site web en le signant avec leur autorité de certification : les navigateurs qui l'incluent n'y verraient que du feu.

page ci-contre

9. Lorsqu'on veut chiffrer une connexion avec un serveur web ou email, on utilise le protocole TLS. C'est un standard qui permet d'encapsuler [page 11] le protocole utilisé habituellement. Par exemple, le protocole web HTTP, quand il est encapsulé dans du TLS, donc chiffré, est appelé HTTPS. Il en va de même pour les protocoles email POPS, IMAPS, et SMTPS.

10. Christopher Soghoian, Sid Stamm, 2011, *Certified Lies : Detecting and Defeating Government Interception Attacks Against SSL*, Financial Cryptography and Data Security [http://files.cloudprivacy.net/ssl-mitm.pdf] (en anglais).

11. Mozilla Foundation, 2014, *Mozilla Included CA Certificate List* [https://www.mozilla.org/about/governance/policies/security-group/certs/included/] (en anglais).

Dans le cas des entreprises, leur but premier n'est pas de certifier des identités mais de gagner de l'argent, en vendant comme service la certification d'identités. Mais vérifier une identité coûte cher. Qu'est-ce qui nous prouve qu'elles le font correctement ? Que leurs clés privées utilisées pour signer sont stockées dans un endroit sûr ? Encore une fois, c'est une question de confiance. On peut espérer que, ne serait-ce que pour maintenir leur activité, ces autorités de certification font bien leur travail…

page 44

Sauf que… des exemples montrent qu'elles le font parfois très mal. Ainsi, en 2008, des chercheurs ont réussi à créer de faux certificats « valides », car six autorités de certifications utilisaient encore des algorithmes cryptographiques qui étaient, de notoriété publique, cassés depuis 2004 [12]. Les certificats ainsi créés sont de « vrais-faux » certificats : le navigateur les reconnait comme vrais, car malgré leur origine frauduleuse, tout laisse à penser qu'ils ont été établis par une autorité reconnue.

En 2011, neuf vrais-faux certificats signés par Comodo, une autorité de certification, ont été créés. Au moins l'un de ces certificats aurait été utilisé sur le web [13]. La société a mis plus d'une semaine à assumer publiquement cette compromission – et nombre d'entre elles ne le font probablement pas dans ce genre de situations, pour éviter la mauvaise publicité [14] et les pertes financières qui vont avec.

Par ailleurs, il semble que si la police ou la justice de leur pays le leur ordonne, certaines autorités de certification donnent aux flics de vrais-faux certificats, établis au nom d'entités qu'ils voudraient surveiller [15]. Cela dit, il faut quand même que ces vrais-faux certificats soient mis en place à l'endroit adéquat sur Internet et combinés à des attaques de l'homme du milieu afin d'être exploités au mieux. Enfin, nos connexions passant en général par plusieurs pays, cette attaque peut tout à fait être déployée par un pays différent de celui depuis lequel on se connecte.

page 64

Dans une brochure commerciale, Packet Forensics, une compagnie américaine qui vend du matériel de surveillance réseau, écrit ainsi que « pour utiliser notre produit dans ce scénario, les utilisateurs gouvernementaux ont la possibilité d'importer une copie d'une clé légitime qu'ils peuvent obtenir (potentiellement grâce à une réquisition judiciaire) » [16]. Le PDG de Packet Forensics aurait confirmé oralement à l'auteur de l'étude que des clients gouvernementaux collaborent avec des autorités de certification pour obtenir des vrais-faux certificats à utiliser lors d'opérations de surveillance [17].

6.4.3 Toile de confiance

Une autre solution à la question de l'authenticité des clés publiques est la toile de confiance, ou *web of trust* en anglais.

Plutôt que de faire confiance à quelques autorités centralisées, il s'agit d'établir un lien de confiance de proche en proche. Ainsi, Betty ne connaît pas Alice, mais elle connaît Daniel, qui connaît Émile, qui connaît Alice. Il y a donc un *chemin de confiance* entre Betty et Alice. S'il n'y avait que ce chemin de confiance, cela impliquerait que Betty place une forte confiance en Émile, qu'elle ne connaît pas directement. Mais Betty

12. Alexander Sotirov, et Al., 2008, *MD5 considered harmful today – Creating a rogue CA certificate* [http://www.win.tue.nl/hashclash/rogue-ca/] (en anglais).
13. Comodo, 2011, *Comodo Fraud Incident* [http://www.comodo.com/Comodo-Fraud-Incident-2011-03-23.html] (en anglais).
14. Jacob Appelbaum, 2011, *Detecting Certificate Authority compromises and web browser collusion* [https://blog.torproject.org/blog/detecting-certificate-authority-compromises-and-web-browser-collusion] (en anglais).
15. Christopher Soghoian, Sid Stamm, 2011, *Certified Lies : Detecting and Defeating Government Interception Attacks Against SSL*, Financial Cryptography and Data Security [http://files.cloudprivacy.net/ssl-mitm.pdf] (en anglais).
16. « To use our product in this scenario, government users have the ability to import a copy of any legitimate key they obtain (potentially by court order) ». Citation extraite du papier de Christopher Soghoian et Sid Stamm cité ci-dessus, et traduite par nos soins.
17. Cette citation se trouve dans une version préliminaire, datant d'avril 2010, du papier de Christopher Soghoian et Sid Stamm cité ci-dessus ; cette version est disponible sur cryptome.org [http://cryptome.org/ssl-mitm.pdf] (en anglais).

connaît aussi Françoise, qui connaît Gaston, qui connaît lui aussi Alice, ainsi que Héloïse, qui connaît Ingrid, qui connaît elle-même Alice. Il y a donc trois chemins de confiance entre Alice et Betty, qui n'a pas besoin d'avoir une confiance totale dans chacune des parties en jeu dans la certification.

Ces toiles de confiance sont couramment utilisées pour l'authentification des logiciels et des communications personnelles, comme des courriers électroniques, en utilisant le standard *OpenPGP*. Elles ne sont hélas pas utilisées couramment pour authentifier des sites web, bien que ce soit possible techniquement [18].

Les toiles de confiance permettent donc de se prémunir des attaques de l'homme du milieu sans devoir faire confiance à des autorités centralisées. Cependant, elles nécessitent de publier des liens entre identités, ce qui a des conséquences qui ne sont pas toujours souhaitables.

page 64

6.5 Confidentialité persistante

Comme on l'a vu, quiconque possède une clé secrète peut l'utiliser pour déchiffrer un texte qui a été chiffré en utilisant la clé publique qui lui est associée. C'est une propriété très utile, mais qui dans certains cas peut se révéler embarrassante.

page 59

Admettons qu'une personne mal intentionnée enregistre une conversation en ligne chiffrée entre deux personnes. Elle ne pourra bien sûr rien lire du contenu de cette conversation dans l'immédiat. Mais elle peut avoir l'idée de s'introduire ensuite chez ces personnes ou dans leur ordinateur et de mettre la main sur leurs clés privées. Dans ce cas, elle sera en mesure de lire, a posteriori, toutes les conversations passées qu'elle aura conservées.

Ce fut le cas il y a quelques années, lorsque les admins du serveur *autistici.org* se rendirent compte lors d'un procès que la police avait mis la main sur les clés secrètes installées sur leur serveur, parce qu'ils produisaient au dossier des échanges d'emails qu'ils n'auraient normalement pas dû être capables de lire [19].

Pour éviter qu'un secret éventé ne compromette a posteriori de nombreux autres secrets qui en dépendent (comme par exemple le contenu de conversations en messagerie instantannée pourtant chiffrées, des échanges de emails, *etc.*) certains logiciels incluent des fonctions dites de confidentialité persistante [20] (ou *Perfect Forward Secrecy*, en anglais).

Elles assurent que même si un jour un secret à long terme, typiquement une clé privée, est découverte par un adversaire, les échanges seront protégés d'une analyse a posteriori.

Dans les faits, au lieu d'utiliser directement la clé publique pour chiffrer les communications, ce type de chiffrement utilise un protocole d'échange de secrets conçu pour fonctionner même sur un canal de communication non sûr, en négociant une clé temporaire à chaque session de communication. La clé secrète d'une paire de clés ne sert, dans ce cas, qu'à s'assurer qu'on communique bien avec la bonne personne, en signant cet échange de secret.

C'est ensuite ce secret temporaire qui est utilisé pour chiffrer de façon symétrique les communications.

tome 1 § 5.3

Une fois la communication terminée, il suffit que les logiciels impliqués oublient ce secret temporaire. Quand bien même quelqu'un mettrait la main sur les clés secrètes

18. Ainsi, le projet Monkeysphere [http://web.monkeysphere.info/] permet d'étendre l'utilisation des toiles de confiance d'OpenPGP à l'authentification de sites web.
19. Austitci, 2005, *CRACKDOWN, violato autistici.org – some legal notes* [http://www.autistici.org/ai/crackdown/legal_en.html] (en anglais).
20. Wikipédia, 2014, *Confidentialité persistante* [https://fr.wikipedia.org/wiki/Confidentialité_persistante]

des deux parties, la confidentialité de la communication ne serait pas compromise : les participants de l'échange eux-mêmes n'y ont plus accès.

6.6 Résumé et limites

La cryptographie asymétrique est donc un bon complément à la cryptographie symétrique dès qu'il s'agit non pas de protéger seulement nos données, mais plutôt le contenu de nos communications : échange d'emails, navigation sur le web, conversations par messagerie instantanée, etc. Son utilisation n'est pas aussi compliquée qu'on pourrait le craindre, et faire du chiffrement une routine permet aux informations particulièrement sensibles d'être noyées dans la masse.

Ce chiffrement est particulièrement efficace lorsqu'il est utilisé de *bout-à-bout*, c'est-à-dire lorsque l'expéditeur d'un message le chiffre de façon à ce que seul le destinataire final puisse le déchiffrer.

tome 1 § 5.1.4

Pour finir ce petit tour des techniques de cryptographie, il est bon de se rappeler que le chiffrement, aussi difficile à casser soit-il, a des limites, qu'on a évoquées dans le premier tome de ce guide. Ces limites touchent notamment à la confiance qu'on met dans l'ordinateur et les logiciels auxquels on confie le chiffrement et le déchiffrement (et donc le texte *en clair*). Elles touchent aussi aux obligations légales de fournir aux autorités les moyens de déchiffrer des communications lorsqu'elles le demandent. Elles touchent enfin à l'évolution de la cryptographie : ce qui est sûr aujourd'hui ne le sera peut-être pas demain.

tome 1 § 5.1.4

Enfin, si le chiffrement permet de cacher le contenu de la communication, les parties impliquées (qui communique avec qui) restent apparentes.

Chapitre 7

Cacher les parties prenantes de la communication : le routage en oignon

Utiliser des protocoles chiffrés permet d'avoir une certaine *confidentialité* sur Internet : un adversaire ne sait pas ce qui se dit. Par contre, un adversaire peut facilement déterminer la source et le destinataire de la communication.

Voyons donc, maintenant, comment et dans quelle mesure on peut essayer de dissimuler d'où vient une communication, et où elle se rend.

7.1 Présentation du routage en oignon

Le routage en oignon, utilisé par exemple par Tor [1], peut fournir un certain *anonymat* sur Internet en masquant d'où provient une communication. En utilisant un tel système, l'adresse IP qui apparaît sur Internet, et qui sera par exemple enregistrée dans les journaux de connexion des serveurs utilisés, n'est pas la nôtre mais celle d'un autre ordinateur.

page 12

7.1.1 Cacher l'origine et la destination

On a vu que les paquets IP, à la manière d'une carte postale, se composent de plusieurs parties. D'une part le contenu, spécifique à chaque application, qui correspond aux données que l'on veut effectivement transmettre : un email, une page web, du son, *etc.* D'autre part, les en-têtes, qui contiennent, entre autres, les adresses IP d'origine et de destination, ainsi que la taille des données transportées. Même en chiffrant les données, les en-têtes restent visibles. Ils révèlent au destinataire de la communication de quelle machine de l'Internet elle provient. Ils révèlent aussi, à tout adversaire qui surveille le trafic échangé, beaucoup de choses sur qui l'on est, voire ce que l'on fait sur Internet.

page 20

Un problème classique concernant l'anonymat est que les destinataires d'une communication peuvent savoir qui en est l'auteur, en regardant les en-têtes. Les intermédiaires autorisés, comme les fournisseurs d'accès à Internet, et parfois des intermédiaires non autorisés, le peuvent aussi. Une forme d'analyse de trafic très simple consiste donc, par exemple, à capturer le trafic entre un expéditeur et un destinataire, et à regarder les en-têtes.

Le chiffrement ne dissimule que le contenu du trafic et non les en-têtes. Il ne protège donc pas contre ce type d'attaques.

De plus, il existe des attaques plus poussées pour trouver la source et la destination d'une communication. Par exemple l'analyse de trafic réseau dont on a parlé

1. L'essentiel de ce qui suit est inspiré du site web de Tor [https://www.torproject.org/overview.html] (en anglais).

page 46 précédemment : un adversaire épie plusieurs points bien choisis de l'Internet (par exemple, la connexion ADSL d'Alice et le serveur qui héberge un blog anonyme auquel elle participe) et compare les motifs de données qui y sont échangés. L'adversaire peut alors confirmer ou infirmer que la communication qu'il surveille vient de telle source et se rend à telle destination.

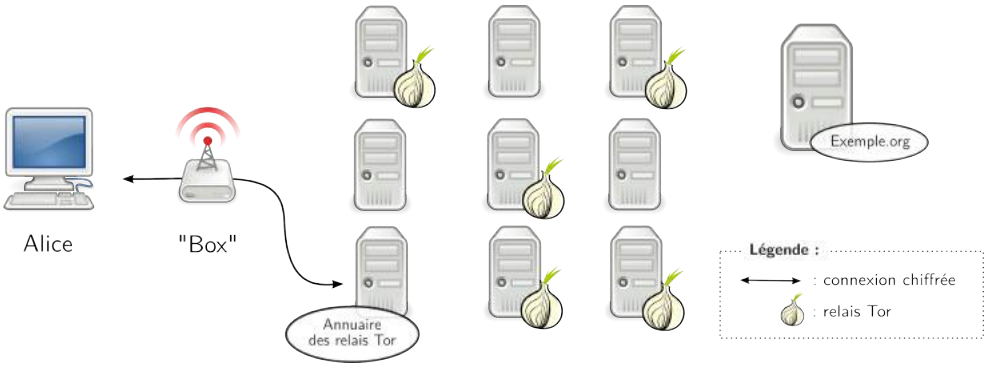
7.1.2 Une solution : un réseau décentralisé d'anonymisation

tome 1 § 4.1 Tor signifie *The Onion Router*, c'est-à-dire « le routage en oignon ». Il s'agit d'un logiciel libre et d'un réseau public qui aide à réduire les conséquences d'une analyse de trafic réseau. Il fait transiter les communications au sein d'un réseau distribué de relais, aussi appelés nœuds, hébergés par des volontaires partout dans le monde. C'est comme utiliser un chemin tortueux et difficile à suivre pour semer un poursuivant, tout en effaçant ses traces à chaque changement de direction. Au lieu d'emprunter un itinéraire direct entre la source et la destination, les paquets de données suivent une trajectoire aléatoire à travers plusieurs relais. Un adversaire ne peut donc pas, en observant un seul point, associer la source et le destinataire.

Création d'un circuit

L'idée, c'est que lorsqu'Alice veut se connecter à exemple.org en utilisant Tor, son ordinateur commence par établir un circuit Tor.

Pour cela, il récupère une liste des nœuds Tor disponibles auprès d'un annuaire :



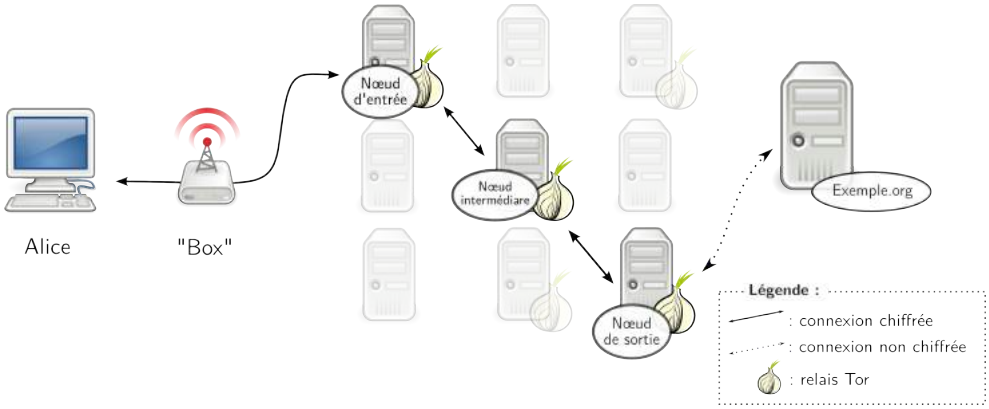
Connexion à un annuaire de relais Tor

Il choisit ensuite un premier relai parmi la liste des relais disponibles, puis établit une connexion à celui-ci. À partir de ce premier relai, il établit une connexion avec un second relai. Enfin, d'après sa liste de nœuds, Tor choisit un nœud de sortie et établit une connexion entre le second relai et ce nœud. Cet ensemble de trois relais constitue ce qu'on appelle un *circuit Tor*. Dès le début de cette phase d'établissement du *circuit Tor*, toutes les communications sont chiffrées.

Utilisation du circuit

Ensuite, les données transiteront successivement par ces trois relais avant d'atteindre le serveur de destination (ici *exemple.org*). La réponse du serveur suivra le même chemin, dans le sens inverse.

Le circuit est parcouru étape par étape, et chaque relai le long du chemin ne connaît que celui qui lui a transmis les données, et celui auquel il va les retransmettre. Aucun relai ne connaît à lui tout seul le chemin complet pris par un paquet de données. Un éventuel intermédiaire ou un relai compromis ne peut pas aisément analyser le trafic réseau pour établir une relation entre la source et la destination d'une connexion. Aucun des ordinateurs ne sait donc que la machine d'Alice se connecte à exemple.org.



Utilisation d'un circuit Tor

Vous noterez qu'un circuit Tor est composé de trois intermédiaires. Si un seul intermédiaire était utilisé, la compromission de celui-ci suffirait à mettre en péril notre anonymat, car cet intermédiaire aurait connaissance à la fois de l'origine d'une communication et de sa destination. Le fait d'utiliser trois relais permet d'éviter ce recoupement sans ralentir la connexion de manière trop importante. page 44

Précaution supplémentaire, le circuit Tor utilisé est modifié automatiquement plusieurs fois par heure.

Chiffrement en oignon

On a vu que l'ordinateur d'Alice négocie une connexion chiffrée avec chaque relai du circuit utilisé. Cela afin que les données qu'elle veut transmettre à exemple.org possèdent plusieurs couches chiffrées. À l'image d'un oignon possédant plusieurs peaux, les données d'Alice seront *enrobées* dans plusieurs couches de chiffrement. La première couche sera chiffrée pour ne pouvoir être lue que par le troisième relai. La seconde, par-dessus la première, sera chiffrée quant à elle pour n'être lisible que du second relai. Enfin, la troisième couche ne pourra être lue que par le premier relai. C'est pour cela que l'on peut parler de *chiffrement en oignon*. À chaque passage par un relai, une couche de chiffrement sera *enlevée*. Aucun des relais ne peut donc déchiffrer les informations qui ne lui sont pas destinées.

Le troisième et dernier relai est appelé « nœud de sortie » : la connexion semblera provenir de lui, il risque donc davantage de se faire ennuyer par les flics.

7.1.3 Les services cachés Tor

Lorsque les utilisateurs de Tor souhaitent également fournir des services, comme par exemple un site web ou un serveur de messagerie instantanée, ils ont la possibilité d'en masquer l'emplacement géographique. C'est ce qui s'appelle un service caché [2].

De la même manière que pour chaque utilisateur du réseau Tor, l'adresse IP du serveur mis en place n'est pas dévoilée. De plus, les personnes souhaitant s'y connecter devront nécessairement utiliser le réseau Tor pour cela. Les services cachés assurent donc un certain anonymat à la fois des serveurs et des clients. Ces services cachés ont des adresses en .onion.

Afin de s'y connecter, les autres utilisateurs de Tor utiliseront le système des « points de rendez-vous » de Tor. Le « point de rendez-vous » est le troisième relai pour chacun des deux protagonistes de l'échange : le client et le service caché. Le client construit

2. The Tor Project, 2013, *Tor : Hidden Service Protocol* [https://www.torproject.org/docs/hidden-services.html] (en anglais).

un circuit Tor avec comme troisième relai ce « point de rendez-vous ». De son côté, le service caché fait de même. Client et service caché se « rencontrent » alors et peuvent échanger des informations.

Ces services cachés peuvent par exemple permettre de mettre en place un site web sur lequel des auteurs publieraient sans craindre la censure. L'identification de l'emplacement physique du serveur qui fournit le site web, comme celle de ses contributeurs et de ses visiteurs, est en effet rendue beaucoup plus difficile que dans le cadre d'un site web conventionnel : elle nécessite de mettre en place une attaque sur le réseau Tor.

page 74

7.2 Participer au réseau Tor

Le réseau Tor repose sur la base du volontariat et est ouvert à tout le monde puisqu'aucun relai ne peut connaître la provenance des communication et leur destination. Et mis à part les nœuds de sortie, aucun relai ne peut connaître le contenu des communications qu'il transporte. Quiconque le souhaite peut donc faire tourner sur la machine de son choix un relai Tor. Ce dernier rejoindra le réseau public et relayera le trafic des personnes utilisant ce réseau.

7.2.1 Monter un relai Tor

Le fait que chaque utilisateur puisse mettre en place un relai introduit de la diversité, renforçant ainsi l'efficacité du réseau Tor dans son ensemble. Cependant, les nœuds ne sont pas égaux devant l'attention qu'ils peuvent attirer. Si un relai placé en première ou seconde position d'un circuit Tor ne peut pas trop être dommageable pour Alice, en France, un nœud de sortie est en revanche plus susceptible d'attirer l'attention sur sa connexion. Les flics pourraient s'intéresser à son relai et éventuellement venir perquisitionner chez elle, voire saisir son ordinateur parce qu'ils font une enquête sur quelqu'un qui est passé par son nœud de sortie pour des activités « suspectes ». Il est bien sûr possible de configurer Tor pour que son relai ne puisse pas être un nœud de sortie, et serve uniquement de premier ou de second relai.

Les relais Tor sont considérés légalement comme des routeurs [3] et, par conséquent, Alice n'est pas tenue de garder des journaux, c'est-à-dire de garder la trace des communications entre une IP et une autre. C'est une bonne chose car, même si un relai Tor pris isolément ne sait pas grand-chose, si le réseau Tor se trouvait, petit à petit, composé en bonne partie de relais qui gardent des journaux, il serait plus facile de redécouvrir les circuits a posteriori.

7.2.2 Monter un bridge Tor

Il est aussi très utile de mettre un place des « bridges » Tor. Il s'agit de relais particuliers qui ne sont pas listés dans les annuaires publics [4] du réseau Tor. Ils peuvent permettre à des utilisateurs dont le fournisseur d'accès à Internet filtre les connexions à Tor de se connecter tout de même au réseau.

7.3 Quelques limites de Tor

Comme tout outil de ce genre, Tor peut facilement donner une fausse impression de sécurité, et faire oublier qu'il répond à un problème précis. S'il répond effectivement plutôt bien au besoin de dissimuler son adresse IP et au besoin de masquer avec quels serveurs on est en communication, il ne résout pas, par contre, un tas d'autres problèmes.

3. Nos Oignons, 2013, *Qu'est-ce que c'est* [https://nos-oignons.net/À_propos/index.fr.html]
4. Il est en revanche possible d'obtenir des adresses de bridges en visitant le site web [https://bridges.torproject.org/] (en anglais).

Comme précisé (en anglais) sur le site de Tor [https://www.torproject.org/], il y a trois choses fondamentales à savoir avant de commencer :

1. Tor ne protège pas si l'on ne l'utilise pas correctement ;
2. Même si l'on configure et que l'on utilise Tor correctement, il y a encore des attaques potentielles qui peuvent compromettre la protection fournie par Tor ;
3. Aucun système d'anonymisation n'est parfait à ce jour, et Tor ne fait pas exception : il serait imprudent de se reposer uniquement sur le réseau Tor si l'on a besoin d'un anonymat strict.

Détaillons à présent quelques-unes de ces limites.

7.3.1 L'utilisateur mal informé ou peu attentionné

Comme souvent, quand on est mal informé, on a de grandes chances de se tromper. Lorsque l'on utilise un outil, a fortiori à des fins de protection de sa vie privée, il est capital de bien comprendre à quoi il sert, mais aussi et surtout à quoi il ne sert pas, tout comme ses différentes limites.

Si Alice utilise régulièrement Tor pour consulter des sites web potentiellement préjudiciables dans son pays, et en même temps ses emails, un adversaire pourrait comparer les journaux du serveur qui héberge sa boîte mail avec ceux des sites web qu'elle a visités. Il est probable alors que ces différentes connexions apparaissent comme venant du même nœud de sortie. En comparant également les heures auxquelles ont lieu les consultations, il est possible d'augmenter d'autant plus les chances de corrélation.

page 29

Dans cet exemple, Alice utilise différentes identités contextuelles en même temps alors que Tor ne prétend pas magiquement séparer celles-ci. Pour éviter de tels recoupements, la solution serait à chercher du côté de la compartimentation des identités contextuelles.

page 54
page 56

7.3.2 Un adversaire voit que l'on utilise Tor

Le fournisseur d'accès à Internet, ou l'administrateur du réseau local d'Alice peut très facilement savoir qu'elle se connecte à un relai Tor, et non à un serveur web ordinaire. [5] En effet, l'adresse IP du serveur vers lequel l'ordinateur d'Alice se connecte sera celui d'un nœud d'entrée du réseau Tor et la liste des nœuds d'entrée est disponible publiquement sur Internet.

Le serveur de destination auquel elle se connecte *via* Tor peut savoir si ces communications viennent d'un nœud de sortie Tor, car la liste de ces nœuds de sortie est également disponible sur Internet.

En utilisant Tor, Alice ne ressemble donc pas à une utilisatrice ordinaire d'Internet. L'anonymat fourni par Tor fonctionne en essayant de mettre tous ses utilisateurs dans le même panier, pour qu'on ne puisse pas les différencier les uns des autres. Plus nombreux seront les internautes utilisant Tor et plus variées seront leurs activités, moins l'utilisation de Tor sera incriminante. La solidité de ce réseau repose notamment sur cet ensemble non distinguable d'utilisateurs, c'est ce qu'on appelle en anglais l'*anonymity set*.

7.3.3 Les nœuds de sortie Tor peuvent espionner les communications qu'ils relaient

Si Tor empêche de savoir où l'on se trouve, il ne chiffre pas les communications en dehors de son propre réseau. Tor ne peut donc pas chiffrer ce qui transite entre le

5. Cette section, ainsi que les suivantes, sont fortement inspirées du site web de *Tails* [https://tails.boum.org/doc/about/warning/index.fr.html]

nœud de sortie et le serveur de destination. Tout nœud de sortie a donc la possibilité de capturer le trafic qui passe par lui.

Par exemple, en 2007, un chercheur en sécurité informatique a intercepté des milliers d'emails privés envoyés par des ambassades étrangères et des ONG à travers le monde en écoutant le trafic sortant du nœud de sortie qu'il administrait [6], en utilisant une attaque de type "homme du milieu".

page 64

Pour se protéger contre de telles attaques, il est nécessaire d'utiliser du chiffrement de bout-à-bout, évoqué dans la partie précédente.

page 59

7.3.4 Attaque de type « motif temporel »

La conception de Tor ne permet pas de protéger contre certains types d'attaques, notamment de l'ordre de l'analyse de trafic [7]. L'attaque par « motif temporel » en est une. L'idée derrière cette attaque est d'observer le rythme d'envoi des données à deux endroits de leur trajet, par exemple sur le premier relai et sur le troisième relai (nœud de sortie) : envoyons par exemple un flux comme du code morse : 3 paquets envoyés en salve, puis 5 secondes de silence, puis 3 paquets, *etc.*

Un adversaire qui voit que l'ordinateur d'Alice envoie sur le premier relai un flux avec un motif temporel donné, et qui observe un flux avec ce même motif sur le nœud de sortie qui va vers exemple.org, peut en déduire que c'est probablement l'ordinateur d'Alice qui est connecté à exemple.org [8].

La force, mais aussi la faiblesse de Tor, c'est que n'importe qui peut l'utiliser, mais aussi avoir un relai Tor : Alice, Betty, une université, la CIA, *etc.* Si un adversaire n'a les informations que d'un seul des relais par lesquels transitent les données, pas de problème. S'il se trouve que par malchance, des adversaires qui coopèrent ont la main sur plusieurs relais, ils peuvent mener une attaque de type « motif temporel ».

Les fournisseurs d'accès à Internet et les gros fournisseurs de contenu ou de ressources utilisées sur de nombreux sites web – encarts publicitaires, fonctionnalités de recherche et de médias sociaux – sont aussi en bonne position pour observer le trafic et donc collaborer à ce type d'attaque.

7.3.5 Tor ne protège pas contre les attaques par confirmation

On vient de voir que la conception de Tor ne permet pas de protéger contre un attaquant qui est capable de mesurer le trafic qui entre et qui sort du réseau Tor. Car si l'adversaire peut comparer les deux flux, il est possible de les corréler *via* des statistiques basiques.

Considérons maintenant un adversaire qui a des raisons de penser que c'est Alice qui publie sur tel blog anonyme. Pour confirmer son hypothèse, il pourra observer le débit du trafic qui sort de la connexion ADSL d'Alice et celui qui entre sur le serveur qui héberge le blog. S'il observe les mêmes motifs de données en comparant ces deux trafics, il pourra être conforté dans son hypothèse.

Tor protège Alice contre un attaquant qui cherche à déterminer qui publie sur le blog anonyme. Mais il ne protège pas contre un adversaire ayant davantage de moyens qui essaye de confirmer une hypothèse en surveillant aux bons endroits dans le réseau puis en faisant la corrélation.

Ce type d'attaque peut aussi s'effectuer avec des hypothèses plus larges. Considérons un adversaire qui a identifié un groupe de connexions ADSL qui l'intéressent, ainsi

6. Kim Zetter, 2007, *Rogue Nodes Turn Tor Anonymizer Into Eavesdropper's Paradise* [http://www.wired.com/politics/security/news/2007/09/embassy_hacks] (en anglais).
7. Wikipédia, 2014, *Attaque par analyse du trafic* [https://fr.wikipedia.org/wiki/Attaque_par_analyse_du_trafic]
8. Voir à ce sujet Wikipédia, 2014, *Tor (réseau)* [http://fr.wikipedia.org/wiki/Tor_(réseau)]

qu'un serveur utilisé à partir de ces connexions. S'il a accès au trafic du groupe de connexions en question, et à celui du serveur, par exemple grâce à une requête légale, l'adversaire peut alors, à partir de cette hypothèse et d'une attaque de type « motif temporel », trouver quelle est la connexion parmi le groupe suspect qui est à l'origine de telle connexion au serveur. Ainsi, un post sur un serveur de blog peut être corrélé à une connexion parmi un groupe de personnes soupçonnées de participer à ce blog anonyme.

7.3.6 Tor ne protège pas face à un adversaire global

Enfin, un cas particulier d'adversaire est celui de l'adversaire global passif. Un adversaire global passif serait une personne ou une entité capable de regarder et donc de comparer le trafic entre tous les ordinateurs d'un réseau. En étudiant, par exemple, les volumes d'informations des différentes communications à travers ce réseau à chaque instant, il serait statistiquement possible d'identifier un circuit Tor car le même flux d'information y apparaîtrait à quelques millisecondes d'intervalle à chaque nœud du circuit. L'adversaire pourraît ainsi relier un utilisateur de Tor et son serveur destinataire.

Un adversaire global, ayant des moyens comparables à ceux de la NSA par exemple, pourrait également mettre en place d'autres attaques visant à briser l'anonymat fourni par le réseau Tor. Cependant, ne pas répondre à une telle menace fait partie des compromis de Tor, et cela pour permettre une navigation raisonnable en termes de délais d'attente, pour le web ou la messagerie instantanée par exemple [9].

Toutefois, les risques résultants de ces limites ne sont pas comparables à ceux rencontrés lors d'une navigation non-anonyme. Tor est l'un des outils les plus efficaces en matière d'anonymat sur Internet, et s'il faut les garder à l'esprit, ces risques ne devraient pas nous détourner de son utilisation avisée.

9. Roger Dingledine, Nick Mathewson, Paul Syverson, 2004, *Tor Project : The Second-Generation Onion Router* [https://svn.torproject.org/svn/projects/design-paper/tor-design.pdf], partie *3. Design goals and assumptions* (en anglais).

Deuxième partie

Choisir des réponses adaptées

La panique s'est désormais emparée de nous. Tout ce qu'on fait sur un ordinateur nous trahit, jour après jour. Qui plus est lorsqu'on croit, à tort, « être en sécurité ».

Mais avant de retourner au pigeon voyageur et à la cache secrète derrière la bibliothèque, qu'on ouvre en tirant sur un faux livre — solutions rustiques à ne pas oublier totalement, ceci dit — il y a un peu de marge. Pas tant que ça, mais tout de même.

C'est cette marge que ce texte s'appliquera dorénavant à cartographier.

Dans cette partie, nous décrirons quelques situations typiques, que nous nommons *cas d'usage*, afin d'illustrer notre propos.

Chapitre 8

Consulter des sites web

8.1 Contexte

On s'intéresse ici à la consultation d'informations disponibles sur le web : lire un périodique, suivre un blog, *etc.* Autant d'activités ordinaires lorsqu'on est *en ligne*.

Cependant, on veut effectuer ces activités de façon discrète, pour diverses raisons, parmi lesquelles on peut citer :

- déjouer la surveillance ou contourner la censure, que ce soit celle d'un patron, d'un proche ou d'un État ;
- éviter la collecte et le recoupement d'informations personnelles à des fins commerciales.

8.2 Évaluer les risques

Ce problème peut paraître trop vaste et complexe pour voir par où le prendre. Découpons-le donc en petits bouts.

8.2.1 Que veut-on protéger ?

Dans ce cas d'usage, ce qui nous importe en premier lieu est l'anonymat, ou tout du moins le pseudonymat : ce qu'on cherche à cacher n'est pas le contenu de *ce qui* est consulté, mais *par qui* il est consulté.

Nous avons vu précédemment que l'utilisation d'Internet, et notamment du web, laisse de nombreuses traces, de diverses natures, à différents endroits ; nombre d'entre elles, telles de petits cailloux, esquissent un chemin qui va de la ressource consultée jusqu'à une maison, un ordinateur, voire la personne qui se trouve derrière. Ce sont donc ces traces sur le réseau, au premier rang desquelles se trouve l'adresse IP, dont on veut se débarrasser. Cependant, l'IP étant nécessaire au bon fonctionnement du réseau, la stratégie sera ici de faire en sorte que les curieux qui suivraient cette piste finissent dans une impasse.

page 12

De plus, on pourra éventuellement vouloir ne laisser aucune trace de notre navigation sur l'ordinateur utilisé, et en particulier sur son disque dur.

8.2.2 De qui veut-on se protéger ?

Cette question est importante : en fonction de la réponse qu'on lui donne, la politique de sécurité adéquate peut fortement varier.

Fournisseur d'accès à Internet

Alice travaille pour une grande entreprise et accède à Internet par l'intermédiaire du réseau de la société. Elle consulte ses blogs préférés sur ses heures de boulot, mais ne souhaite pas que son employeur le sache.

Dans ce cas, Alice souhaite se protéger de l'œil indiscret de son fournisseur d'accès à Internet, en l'occurrence son entreprise. L'adversaire a ici accès à l'ensemble du trafic réseau qui transite par sa connexion pour lequel il joue le rôle de facteur. Il n'a, par contre, pas d'yeux placés en d'autres points d'Internet.

Fournisseurs de contenu

Betty est inscrite sur un forum de la police nationale, et passe — non sans un malin plaisir — un certain temps à semer la zizanie dans les discussions entre flics.

Dans ce cas, Betty ne souhaite pas rendre transparent au site hébergeant le forum qu'elle est la fauteuse de troubles. Comme vu précédemment, son adresse IP sera conservée plus ou moins longtemps par le site visité. Dans ce cas-ci l'adversaire aura accès aux en-têtes IP, ainsi qu'aux en-têtes HTTP car il en est le destinataire.

page 36
page 12

Adversaires divers et variés

page 29

Agathe va régulièrement consulter le site de publication de documents confidentiels sur lequel Benoît a publié des relevés bancaires. Le sujet étant sensible, elle sait pertinemment que le blog en question pourrait être surveillé. Elle ne veut donc pas qu'on sache qu'elle va le consulter.

page 5

L'adversaire ici n'a pas de place fixe sur le réseau, il peut se situer au niveau de l'ordinateur d'Agathe, au niveau de sa « box », au niveau du blog ou bien à tout autre endroit sur le chemin entre son ordinateur et le blog. L'adversaire peut également se situer à plusieurs endroits en même temps.

page 18

8.3 Définir une politique de sécurité

tome 1 ch. 7

Posons-nous maintenant les questions exposées dans notre méthodologie :

1. Quel ensemble de pratiques, d'outils nous protégeraient de façon suffisante contre nos adversaires ?
2. Face à une telle politique de sécurité, quels sont les angles d'attaque les plus praticables ?
3. Quels sont les moyens nécessaires pour les exploiter ?
4. Pensons-nous que ces moyens puissent être utilisés par nos adversaires ?

8.3.1 Première étape : demander à ceux qui voient

page 36

Angle d'attaque le plus praticable pour l'adversaire : analyser les données enregistrées par les serveurs hébergeant les ressources consultées.

Moyens nécessaires :

- se connecter au serveur qui fournit la connexion si l'adversaire est le fournisseur d'accès à Internet, ou collabore avec lui ;
- se connecter au serveur qui héberge la ressource si l'adversaire est, ou collabore avec, le fournisseur de contenu.

Si l'adversaire est le fournisseur d'accès Internet ou le fournisseur de contenu, il lui suffira de consulter ses journaux de connexions. Mais il est également possible à d'autres adversaires d'accéder à ces informations, par le biais d'une requête légale,

page 36
page 39

d'un contrat commercial, d'une collaboration volontaire[1], voire d'un piratage. page 44

Crédibilité d'une telle attaque : probable si notre connexion ou le site visité attirent l'attention de l'adversaire.

Contre ce type d'attaque, une solution efficace est d'utiliser le routage en oignon[2] en utilisant le réseau Tor selon des modalités qu'on présentera plus loin. Pour s'assurer un maximum d'anonymat, il sera alors nécessaire de ne pas mélanger ses activités quotidiennes normales avec celles que l'on souhaite plus discrètes, afin ne pas créer de liens entre nos différentes identités contextuelles. page 69 page 53

8.3.2 Deuxième étape : regarder sur l'ordinateur utilisé

Lorsque nous utilisons Tor, l'adversaire observant les données circulant sur le réseau ne peut pas savoir à la fois d'où viennent et où vont ces données, et doit alors trouver un autre moyen d'y parvenir.

Angle d'attaque le plus praticable : avoir accès aux traces laissées sur l'ordinateur par les sites visités. tome 1 ch. 2

Moyens nécessaires : accéder à l'ordinateur utilisé.

Crédibilité d'une telle attaque : dans le cas d'Alice qui utilise l'ordinateur de son boulot, cela est très facile pour son adversaire. Dans d'autres cas et selon l'adversaire, cela nécessite soit un cambriolage (également appelé perquisition, quand il est légal), soit de corrompre l'ordinateur cible de l'attaque, par exemple pour y installer un logiciel malveillant. tome 1 ch. 3

Pour se prémunir contre cette attaque, il est nécessaire de chiffrer son disque dur pour rendre difficiles d'accès les traces laissées. Ou, mieux encore, éviter, dès le départ, de laisser des traces : en utilisant un système *live* amnésique, qui n'enregistrera rien sur l'ordinateur utilisé. tome 1 ch. 15 tome 1 ch. 14

8.3.3 Troisième étape : attaquer Tor

Angle d'attaque : exploiter les limites de l'anonymat fourni par Tor, par exemple en effectuant une attaque par confirmation. page 72 page 74

Moyens nécessaires : être capable de surveiller plusieurs points du réseau, par exemple la connexion utilisée *et* le site consulté.

Crédibilité d'une telle attaque : un adversaire comme une entreprise qui cherche à surveiller ses salariés a peu de chances de monter une telle attaque. Idem pour les gendarmes de Saint-Tropez. Elle peut cependant être à la portée d'un fournisseur de services réseau d'envergure nationale ou mondiale, voire de flics spécialisés. Encore une fois, n'oublions pas qu'il y a une différence notable entre « avoir la capacité technique de mettre en place une attaque » et « mettre effectivement en place une telle attaque ». Cette différence peut notamment tenir au coût économique, au retour sur investissement, d'une telle attaque ciblée.

Rappelons au passage que de nombreuses autres attaques contre Tor sont possibles ou envisagées. Retenons surtout qu'il est nécessaire de bien comprendre les objectifs et les limites du routage en oignon pour ne pas se tirer une balle dans le pied. page 70 page 72

1. Jacques Follorou, Le Monde, 2014, *Espionnage : comment Orange et les services secrets coopèrent* [http://www.lemonde.fr/international/article/2014/03/20/dgse-orange-des-liaisons-incestueuses_4386264_3210.html].

2. Contre certains des adversaires listés ici, des solutions techniques moins poussées que le routage en oignon peuvent suffire. L'utilisation d'un VPN [https://fr.wikipedia.org/wiki/Réseau_privé_virtuel] par exemple. Cependant, qui peut le plus peut le moins, et Tor protège contre beaucoup plus d'attaques possibles qu'un VPN, qui n'insère entre nous et la ressource consultée qu'un seul intermédiaire.

8.4 Choisir parmi les outils disponibles

En fonction de nos besoins et de notre politique de sécurité, il nous faudra choisir parmi différents outils.

8.4.1 Tor Browser Bundle ou Tails

Tor Browser Bundle

Configurer un navigateur web pour utiliser Tor correctement est un exercice difficile. C'est notamment pour pallier à cette difficulté qu'existe le *Tor Browser Bundle (TBB)*. Le TBB est un *pack* de logiciels : il fournit un navigateur web préconfiguré pour surfer de façon anonyme, en utilisant le réseau Tor, à partir de notre système d'exploitation habituel[3]. Une fois le *Tor Browser Bundle* installé, on peut choisir d'utiliser ce navigateur web utilisant Tor, ou notre navigateur web habituel.

page 111

Avantages Le *Tor Browser Bundle* permet de naviguer sur le web avec Tor depuis notre système d'exploitation habituel. Il permet par exemple de travailler sur un document avec nos outils habituels, tout en cherchant des informations sur le web de façon anonyme.

Inconvénients Le *Tor Browser Bundle* s'exécutant sur le système d'exploitation habituel, cela implique qu'une faille dans celui-ci permettrait à un adversaire de contourner la protection offerte par l'usage du réseau Tor. Mais surtout, utilisé en dehors d'un système amnésique, le TBB laissera probablement des traces sur le disque dur de l'ordinateur utilisé.

Ensuite, le TBB n'est pas disponible dans les dépôts de paquets Debian. Il faudra donc se charger manuellement de vérifier son authenticité et de le garder à jour, pour corriger des problèmes liés à la sécurité.

De plus, lors de mises à jour de Firefox, sur lequel le TBB est basé, il peut y avoir un délai plus ou moins long d'ici à ce que ces mises à jour soient prises en compte dans le TBB. Pendant ce délai, il présentera des failles de sécurité connues, et publiées.

Enfin, avec les réglages par défaut, par ailleurs modifiables, il peut arriver de perdre définitivement toutes ses lectures et recherches en cours si jamais le TBB vient à planter par exemple.

D'autre part, le TBB n'empêche pas d'autres programmes de se connecter à Internet sans passer par Tor, et ce même s'ils sont ouverts depuis le navigateur du TBB (logiciels P2P, afficheurs de fichiers PDF, lecteurs multimedia, *etc.*)

Tails

Tails[4] est un système *live* dont le but est de préserver la confidentialité et l'anonymat de ses utilisateurs. Il permet d'utiliser Internet de manière anonyme quasiment partout et depuis n'importe quel ordinateur. De plus, il ne laisse aucune trace des activités effectuées sur l'ordinateur, à moins qu'on ne le lui demande explicitement.

tome 1 ch. 14

Avantages En utilisant *Tails*, non seulement on ne laisse pas de trace sur l'ordinateur utilisé, mais les logiciels ayant besoin d'accéder à Internet sont configurés pour passer par le réseau Tor, et les connexions directes (qui ne permettent pas l'anonymat) sont bloquées.

3. Dans notre cas, il s'agit de Debian, mais le *Tor Browser Bundle* fonctionne aussi avec n'importe quelle autre distribution GNU/Linux, tout comme avec Windows ou Mac OS.
4. Site de Tails [https://tails.boum.org/index.fr.html]

De plus, comme il s'agit d'un système *live*, *Tails* démarre à partir d'un DVD, d'une clé USB ou d'une carte SD, sans modifier le système d'exploitation installé sur l'ordinateur. Il peut donc être utilisé autant à la maison que chez un ami, ou à la bibliothèque du coin.

Pour plus d'informations, consultez la page « À propos » de *Tails* [https://tails.boum.org/about/index.fr.html].

Inconvénients Tout d'abord, *Tails* étant un système d'exploitation à part entière, il est nécessaire, pour l'utiliser, de redémarrer l' ordinateur [5]. Il est aussi plus complexe à installer que le *Tor Browser Bundle*. Enfin, il est nécessaire d'avoir sur soi une clé USB ou une carte SD (d'une capacité d'au moins 4 GB) ou bien un DVD, contenant *Tails*.

tome 1 § 1.4.1

Ensuite, du fait de l'amnésie du système, si jamais le navigateur web vient à planter, on perd toutes les pages que nous étions en train de consulter, tout comme dans le cas du TBB.

Pour ne pas mélanger ses activités quotidiennes normales avec celles que l'on souhaite plus discrètes lorsqu'on utilise *Tails*, il est nécessaire de redémarrer sa machine quand on passe d'une identité contextuelle à une autre.

Au chapitre des inconvénients inhérents à *Tails*, il y a aussi le délai entre les mises à jour (de sécurité) de programmes par ailleurs inclus dans *Tails*, et les mêmes mises à jour de ces logiciels dans *Tails*. Cet inconvénient est similaire à celui du TBB concernant le délai entre les mises à jour de Firefox et leur prise en compte dans le TBB.

Pour plus d'information, se reporter à la page « Avertissements » de *Tails* [https://tails.boum.org/doc/about/warning/index.fr.html].

8.4.2 Faire son choix

On doit en fin de compte faire son choix entre :

- utiliser son système d'exploitation habituel ;
- utiliser un système *live* amnésique.

En d'autres termes, quelles traces (éventuellement chiffrées) est-on prêt à laisser sur l'ordinateur, la clé USB ou la carte SD utilisés ? A-t-on besoin du reste de son environnement lors de la navigation anonyme ?

Encore une fois, il n'y a pas de bonne ou de mauvaise réponse : il s'agit de choisir la solution qui nous convient le mieux. De plus, il est tout a fait possible de tester une solution puis de passer à une autre si nécessaire.

Au final, les deux possibilités suivantes s'offrent à nous :

- utiliser le *Tor Browser Bundle* depuis une Debian chiffrée. Cela permet de naviguer de manière anonyme tout en utilisant son système habituel. Par contre, des traces (chiffrées) seront probablement laissées sur le disque dur de l'ordinateur ;
- utiliser le navigateur web de *Tails*. On ne laisse pas de traces sur le disque dur de l'ordinateur utilisé, voire pas de traces du tout si l'on n'utilise pas la persistance ;

tome 1 ch. 15

tome 1 § 14.5

Une fois votre choix effectué, consultez ci-dessous le paragraphe correspondant.

5. On peut aussi utiliser *Tails* dans une machine virtuelle [tome 1 ch. 22] dans le système utilisé habituellement. Dans ce cas, la mémoire de la machine virtuelle sera visible pour celui-ci, et toutes les données utilisées, mots de passe compris, seront à la portée d'une faille de programmation ou d'un éventuel logiciel malveillant. De plus, si celui-ci utilise de la swap [tome 1 § 1.5.4], il est possible que des données de la machine virtuelle finissent par être écrites sur le disque dur. L'amnésie du système *Tails* utilisé de cette façon est donc quasiment impossible à garantir.

8.5 Naviguer sur des sites web avec le Tor Browser Bundle

Si, après avoir pesé le pour et le contre, on décide d'utiliser le *Tor Browser Bundle* plutôt que *Tails*, certaines précautions sont bonnes à prendre.

8.5.1 Préparer sa machine et installer le Tor Browser Bundle

tome 1 ch. 8

Tout d'abord, comme nous n'utilisons pas un système *live*, des traces de navigation (cookies, fichiers téléchargés…) seront inscrites sur notre disque dur. Appliquer la même politique que pour un nouveau départ est une bonne piste. Ensuite il faut télécharger et installer le *Tor Browser Bundle* correctement. Le chapitre expliquant comment installer le TBB décrit cette procédure.

page 111

8.5.2 Utiliser le Tor Browser Bundle

page 111

Dans la page concernant l'installation du TBB, il est également expliqué comment le démarrer. Cet outil est spécialement conçu pour être le plus simple possible à utiliser. Au moment de son lancement, tous les logiciels dont nous avons besoin (Tor et le navigateur web Firefox) démarreront et seront paramétrés. Il suffira donc d'attendre que la fenêtre de Firefox s'ouvre et nous pourrons commencer la navigation *via* le réseau Tor.

Attention : seule la consultation de sites web *via* cette fenêtre de navigateur garantit une connexion anonymisée. Toutes vos autres applications (client mail, messagerie instantanée, *etc.*) laisseront apparaître votre véritable adresse IP.

page 12

De plus, une fois cette fenêtre fermée, il vous faudra relancer le *Tor Browser Bundle* et attendre qu'une nouvelle fenêtre de Firefox s'ouvre pour reprendre une navigation qui passe par le réseau Tor.

8.5.3 On perçoit vite les limites

Le *Tor Browser bundle* est un très bon outil de par son utilisation simplifiée, mais on en perçoit vite les limites. En effet, seules les connexions initiées par le TBB passent par le réseau Tor. Si l'on veut utiliser un autre navigateur, la connexion ne passera alors plus par ce réseau, ce qui peut être fâcheux. En cas d'inattention on peut donc vite se tromper de navigateur et penser que notre navigation passe par le réseau Tor alors que ce n'est pas le cas… De plus, il ne permet pas d'utiliser Tor pour autre chose que naviguer sur le web.

De plus, l'anonymat d'une connexion ne tient pas seulement à la falsification de l'adresse IP. Toutes les traces que nous laissons sur le web *et* sur notre ordinateur peuvent nous trahir un jour ou l'autre, et le TBB ne protège pas contre cela.

Enfin, avec le TBB, il est plus facile de finir par mélanger des identités contextuelles, d'autant plus qu'il est utilisé dans le même environnement que celui de l'identité principale de la personne qui l'utilise.

8.6 Naviguer sur des sites web avec Tails

8.6.1 Obtenir et installer Tails

tome 1 § 4.1

Tails est un logiciel libre, et peut donc être téléchargé, utilisé et partagé sans restriction. Il fonctionne sur un ordinateur indépendamment du système installé. En effet, Tails se lance sans utiliser le disque dur, depuis un support externe : un DVD, une carte SD ou une clé USB suffisent.

tome 1 § 14.2.1

Après avoir téléchargé *Tails*, il nous faudra vérifier l'image ISO afin de s'assurer que le téléchargement s'est bien déroulé.

tome 1 § 14.2.2

Une fois la vérification effectuée, on peut procéder à l'installation sur une clé USB,

tome 1 § 14.2.3

une carte SD ou un DVD.

8.6.2 Démarrer Tails

Maintenant que l'on a installé *Tails*, on peut redémarrer et commencer à l'utiliser, sans altérer le système d'exploitation présent sur l'ordinateur.

tome 1 § 14.4

8.6.3 Se connecter à Internet

Une fois le démarrage de *Tails* achevé, c'est-à-dire une fois que le bureau a terminé de s'afficher, il ne nous reste plus qu'à nous connecter à Internet afin d'aller naviguer sur le web.

page 115

8.6.4 Limites

Une telle solution repose sur l'utilisation de Tor et de *Tails*, et hérite donc des limites de ces deux outils :

Concernant les limites de Tor, elles ont été évoquées précédemment dans le paragraphe « Troisième étape : attaquer Tor ».

page 81

Pour les limites de *Tails*, vous trouverez une liste approfondie d'avertissements sur le site web du projet [https://tails.boum.org/doc/about/warning/index.fr.html].

Nous ne pouvons que vous inviter à lire et relire attentivement ces deux documents.

Chapitre 9

Publier un document

9.1 Contexte

Après avoir terminé la rédaction d'un document sensible, on souhaite le publier sur Internet tout en conservant notre anonymat (le fait qu'il ne puisse être associé à aucun nom) ou notre pseudonymat (le fait qu'il ne puisse être associé qu'à un nom choisi et différent de notre nom civil) .

tome 1 ch. 9

En prime, on voudrait pouvoir y inclure une adresse de contact public correspondant à ce pseudonyme.

9.2 Évaluer les risques

9.2.1 Que veut-on protéger ?

Le contenu du document est public. On ne s'intéresse donc pas à sa confidentialité. Par contre, on cherche à cacher les liens entre le document et les personnes qui l'ont rédigé. C'est donc ici l'**anonymat**, ou le **pseudonymat**, qui nous intéresse.

9.2.2 Contre qui veut-on se protéger ?

Comme dans le cas d'usage précédent, nous chercherons ici à nous protéger des regards indiscrets qui chercheraient à savoir *qui* fait *quoi* sur le web.

page 79

On fera d'autant plus attention aux traces laissées qu'il s'agit justement ici de publier un document dont on suppose qu'il peut déplaire à une ou plusieurs personnes ayant un certain pouvoir de nuisance. Il est alors probable que débute une recherche d'indices pour tenter de retrouver le ou les auteurs du document, par exemple en adressant des requêtes légales à l'hébergeur.

page 39

page 21

9.3 Définir une politique de sécurité

Nous allons traiter successivement la publication de documents puis l'utilisation d'un contact public lié à ceux-ci.

9.3.1 Publication

Publier un document revient techniquement à « sauvegarder » celui-ci sur un serveur connecté à Internet, que l'on appelle l'**hébergeur**. On passe souvent par un site web pour réaliser cette opération. Cependant, on ne va pas utiliser les mêmes sites si l'on veut publier du texte, du son ou de la vidéo.

page 21

page 22

Il s'agit donc de bien choisir notre hébergeur en ayant à l'esprit les nombreux critères entrant en jeu : type de document, disponibilité, conditions d'hébergement, résistance

de l'hébergeur aux pressions judiciaires, risques que notre document fait courir à celui-ci, *etc.* Un liste plus exhaustive de ces critères est disponible page 117 dans la partie « Outils ».

Une fois notre choix effectué, il va falloir être sûr que notre document reste consultable : en effet, si notre publication ne plaît pas à notre hébergeur, qu'il reçoit des pressions, voire une requête légale exigeant sa suppression, notre œuvre pourrait devenir indisponible.

Pour éviter ce genre de désagréments, on peut multiplier les hébergements d'un même fichier, si possible sur des serveurs situés dans différents pays. La mise en ligne d'un fichier étant beaucoup plus rapide qu'un recours judiciaire, cela semble être une bonne solution pour éviter la censure.

Quels seront alors les angles d'attaque à la portée d'un éventuel adversaire ?

9.3.2 Première étape : c'est écrit en bas à gauche

L'adversaire dispose de prime abord d'un gros volume de données au sein duquel chercher des traces : le contenu du document.

Ainsi, une éventuelle signature comme un pseudonyme ou une ville, une date, la langue dans laquelle le document est écrit, voire tout simplement le thème du document sont autant d'indices qui peuvent mener à ses auteurs. Un texte qui décrit les pratiques abusives de la société Machinex en novembre 2012 a probablement été rédigé par des employés de cette société ou des gens qui partageaient leur lutte à cette date.

page 54 L'adversaire peut aussi tenter une analyse stylométrique pour le comparer à d'autres textes, anonymes ou non, et essayer d'en déduire des informations sur les auteurs. À notre connaissance, ce type d'attaque n'est réellement effective que lorsqu'on a déjà de forts soupçons sur un sous-ensemble d'auteurs potentiels, mais c'est un champ de recherche récent. Vu que l'on souhaite diffuser largement ce document, on ne pourra pas masquer le contenu. Cependant, si l'on pense nécessaire de s'en donner la peine, on pourra avoir une attention particulière à changer son style d'écriture.

tome 1 § 2.6 Enfin, si l'on publie notre document sans prendre de plus amples précautions, un adversaire peut chercher d'éventuelles métadonnées qui lui fourniraient quelques informations.

Ces différentes méthodes ne demande pas de grandes compétences techniques et sont donc à la portée de beaucoup d'adversaires.

Pour s'en protéger, on suivra les recettes suivantes :

tome 1 ch. 9
- si possible, on travaillera sur notre document en utilisant dès le début des méthodes limitant les métadonnées qui pourront être enregistrées ;

tome 1 ch. 24
- dans tous les cas, il est bon de supprimer d'éventuelles métadonnées avant publication.

9.3.3 Deuxième étape : demander à ceux qui voient

En l'absence de traces facilement exploitables à l'intérieur du document, l'un des angles d'attaque le plus praticable est alors de chercher les traces de sa publication sur le réseau.

page 39 Selon ses pouvoirs, notre adversaire peut effectuer une requête légale auprès de l'hébergeur du contenu ou trouver une autre façon de se procurer ses page 36 journaux de connexion et ainsi obtenir l'adresse IP utilisée. Il peut ensuite se tourner vers le FAI page 38 correspondant à cette adresse IP pour avoir le nom de l'abonné.

Ici aussi, pour y faire face, on utilisera Tor pour se connecter à Internet en brouillant cette piste avant de publier notre document.

Quant au choix de l'hébergement, les questions discutées ci-dessus s'appliquent toujours. De plus, certaines des plateformes sur lesquelles on voudrait déposer notre document sont susceptibles de ne pas fonctionner si Tor est utilisé, ou d'utiliser des technologies comme Flash qui sont fortement déconseillées lorsque l'on souhaite conserver son anonymat : cela restreindra les hébergeurs utilisables. page 26

> Il est aussi possible d'héberger nous-mêmes notre document grâce aux *services cachés* de Tor [page 71] : ils permettent de rendre disponible un serveur web ou un autre type de serveur sans avoir à révéler son adresse IP. Ils n'utilisent pas d'adresse publique et peuvent donc fonctionner aisément même derrière un pare-feu [page 20] ou une autre *box* faisant de la traduction d'adresse réseau (NAT) [page 18]. La procédure est un peu complexe. On ne la documentera pas en détail dans cette édition du guide, mais une traduction française du site web de Tor [`http://tor.hermetix.org/docs/tor-hidden-service.html.fr`] est une bonne base pour tenter l'aventure.

Pour publier notre document, on commencera en pratique par suivre la recette trouver un hébergement web. page 117

Dans la plupart des cas, la publication se fera grâce à un navigateur web. On suivra donc la piste « navigateur web » du cas d'usage précédent. page 84

Il faudra alors très certainement ajouter un certificat ssl pour pouvoir accéder au site de l'hébergeur en utilisant une connexion chiffrée. page 121

9.3.4 Troisième étape : regarder sur l'ordinateur utilisé

Cet angle d'attaque est similaire à celui décrit dans la section « regarder sur l'ordinateur utilisé » du cas d'usage précédent. Nous ne pouvons donc que vous inviter à aller lire (et relire) ce chapitre. page 81

9.3.5 Quatrième étape : attaquer Tor

En désespoir de cause, l'adversaire peut aussi tenter d'attaquer Tor (voir la section « attaquer Tor » du cas d'usage précédent). page 81

9.4 Contact public

Lorsqu'on publie un document, on peut vouloir être contacté par les personnes qui vont nous lire. Ce contact ouvre de nouvelles possibilités d'attaques à un adversaire en quête de failles à exploiter.

Si l'on a pris toutes les précautions afin d'être le plus anonyme possible lors de la publication du document, mais que notre adresse de contact est nom.prénom@exemple.org, ces précautions seront sans intérêt : l'adresse de contact donne directement à l'adversaire notre nom.

Pour éviter cette erreur, on veillera donc à avoir un pseudo qui sera utilisé uniquement pour ce document ou un groupe de documents en fonction de l'identité contextuelle que l'on souhaite adopter. page 53 page 54

L'adversaire cherchera alors à savoir qui se cache derrière ce pseudonyme. Pour tenter de masquer « qui utilise cette adresse email », le cas d'usage "Envoyer des emails sous page 96

un pseudonyme" pourra nous aider.

page 97 Enfin, on pourrait avoir envie de cacher le contenu des emails échangés, mais ceci peut apparaître très complexe : dans la mesure où l'on souhaite avoir une adresse de contact publique, l'*accessibilité* peut rentrer en conflit avec la discrétion.

On peut ainsi prendre tout un ensemble de précautions pour augmenter l'anonymat de notre contact, mais l'on peut difficilement agir sur l'autre « bout du tuyau ». Les personnes qui vont nous contacter peuvent alors prendre des risques en dialoguant avec nous, sans penser à leur anonymat. Rappeler et expliciter les conditions de confidentialité et d'anonymat est alors indispensable. De plus, on ne sait jamais vraiment qui nous contacte, il faudra alors faire attention à ce que l'on raconte si l'on ne veut pas se compromettre.

Chapitre 10

Échanger des messages

10.1 Contexte

On souhaite maintenant échanger des messages avec d'autres personnes, que ce soit pour souhaiter une bonne année à mamie, ou pour travailler sur un document sensible. On ne se soucie pas de la synchronicité de l'échange, à l'inverse d'une conversation téléphonique ou d'un dialogue en messagerie instantanée : on parle dans ce cas de communication *asynchrone*.

tome 1 ch. 9

Un autre cas d'usage sera consacré au dialogue synchrone. Concentrons-nous plutôt, pour l'instant, sur le courrier électronique, ou email.

page 101

10.2 Évaluer les risques

10.2.1 Que veut-on protéger ?

Lorsqu'un courrier électronique est envoyé, diverses informations sont potentiellement dévoilées à nos adversaires. Lesquelles ?

Quand on se pose cette question, c'est bien souvent le *contenu* du message qui vient à l'esprit en premier lieu. Si tous les messages que nous échangeons ne sont pas nécessairement top-secrets, certains méritent plus de discrétion que d'autres : afin d'éviter que les détails de nos relations intimes soient étalés, ou encore car le contenu d'un message peut nous attirer des ennuis, allant de la perte d'un boulot à un séjour en prison. Plus généralement, nous ne débordons pas d'enthousiasme à l'idée que le facteur puisse lire aujourd'hui toutes les lettres qu'on a reçues ces dernières années, pour se mettre en bouche, avant d'attendre avidement celles qui arriveront demain. Pourtant, lorsqu'on échange du courrier électronique sans précautions particulières, les intermédiaires peuvent lire nos communications de façon totalement transparente, comme s'il s'agissait de cartes postales.

Au-delà du contenu de ces cartes postales, il peut être intéressant de masquer les informations contextuelles, telles que la date de l'échange, les identités des protagonistes, leurs localisations, *etc.* qui peuvent être révélées par exemple dans les en-têtes HTTP, les en-têtes des emails, ou dans le corps du message lui-même.

page 29
page 30

Le fait qu'une certaine personne écrive à telle autre peut constituer en soi une information sensible. En effet, il arrive que ce soit les relations entre des gens qui soient visées par certaines formes de surveillance, afin de reconstituer un réseau d'opposants politiques [1] par exemple. Ces traces persisteront généralement dans les en-têtes des emails et les journaux de connexion.

page 31
page 30
page 36

1. Jean-Marc Manach, 2011, *Réfugiés sur écoute* [http://owni.fr/2011/12/01/amesys-bull-eagle-surveillance-dpi-libye-wikileaks-spyfiles-kadhafi/]

10.2.2 Contre qui veut-on se protéger ?

On peut vouloir dissimuler tout ou partie de ces informations aux diverses machines qui peuvent y avoir accès, ainsi qu'aux personnes ayant accès à ces machines.

page 29

Parmi ces machines, viennent tout d'abord les serveurs impliqués. Au minimum, pour un message envoyé par Alice (`alice@exemple.org`) à Betty (`betty@fai.net`), il s'agira :

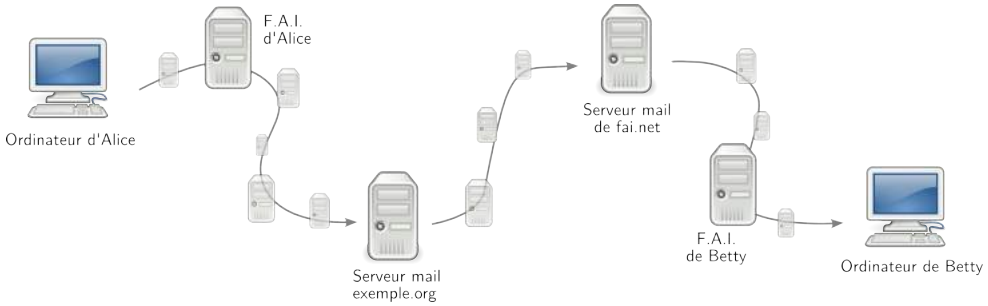
- du serveur chargé par Alice d'envoyer le message : généralement, ce sera `exemple.org` ;
- du serveur chargé de recevoir un message et de le stocker dans la boîte mail de Betty : `fai.net`.

page 15

Mais ce n'est pas tout. De nombreux autres ordinateurs (les *routeurs*) sont situés le long du trajet, et ont accès à l'information qu'ils transportent :

page 28

- entre l'ordinateur d'Alice et son FAI ;
- entre le FAI d'Alice et son serveur mail `exemple.org` ;
- entre `exemple.org` et le serveur mail de Betty `fai.net` ;
- lorsque Betty consultera sa boîte mail, le message cheminera entre le serveur mail `fai.net` et son FAI,
- enfin, entre le FAI de Betty et son ordinateur.



Un mail transite par de nombreux intermédiaires

Les admins de ces machines sont les premiers à avoir accès aux informations que celles-ci traitent, mais n'en ont pas forcément l'exclusivité. Ces informations peuvent se retrouver aux mains de pirates plus ou moins gouvernementaux, munis ou non de requêtes légales.

page 44

page 39

Pour finir, chaque consultation d'une boîte mail, chaque envoi de message, est susceptible de laisser des traces sur l'ordinateur utilisé. Il peut être pertinent de dissimuler celles-ci aux curieux qui seraient en mesure de jeter un œil au contenu de nos disques durs.

tome 1 ch. 2

10.3 Deux problématiques

On peut avoir comme souci de protéger à la fois notre identité - voire celles de nos destinataires - et le contenu des échanges. Il s'agit donc des informations contenues dans les deux parties de notre carte postale numérique, à gauche le texte, à droite les en-têtes. Ces informations apparaissent tout au long du parcours de nos messages et peuvent être la cible d'attaques. La politique de sécurité que l'on va définir va notamment dépendre de la façon dont nous consultons nos emails. En effet, son utilisation peut impliquer divers protocoles qui n'ont pas les mêmes conséquences en termes de traces.

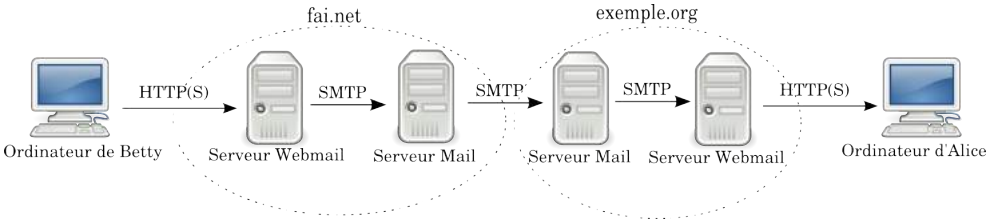
page 20

10.4 Webmail ou client mail ?

Il existe deux manières d'utiliser l'email, permettant les mêmes actions : l'utilisation du webmail ou d'un client mail. Ce choix repose sur différents critères, sachant que les deux sont utilisables pour une même adresse email, et que le choix de l'un ou de l'autre n'est pas irréversible.

10.5 Webmail

Un **webmail** est un site web permettant de consulter ses emails *via* un navigateur web. Son usage s'est répandu comme une traînée de poudre depuis le début des années 2000, à tel point qu'on en aurait presque oublié les autres manières de faire de l'email. *Hotmail* et *Gmail* sont des exemples très populaires d'hébergeurs qui favorisent son utilisation (même s'ils ne sont pas utilisables qu'en webmail). Ici encore, on a affaire à une tendance du web 2.0, plus besoin d'avoir son système d'exploitation pour accéder à sa boîte mail (que ce soit sur son ordinateur ou sur la clé USB contenant un système *live*) : un accès à Internet suffit. page 49



Alice et Betty utilisent un webmail

Le webmail c'est en fin de compte une interface web qui nous permet d'agir sur des serveurs mails. Schématisons un échange d'email entre Alice et Betty, qui utilisent toutes deux le webmail :

- le « chemin réseau » entre l'ordinateur de Betty et sa boîte mail hébergée par `fai.net` sera parcouru *via* un protocole web (HTTP ou HTTPS)
- s'ensuivra un petit bout de chemin chez `fai.net` qui assurera le passage du webmail à l'email
- suivi d'un voyage en protocole email (SMTP) entre `fai.net` et `exemple.org`
- de nouveau un petit bout de chemin, chez `exemple.org` cette fois-ci, entre protocole email et web
- puis du web (HTTP ou HTTPS) jusqu'à l'ordinateur d'Alice.

10.5.1 Avantages

Parmi les avantages du webmail, de même que pour chaque application web, on peut noter l'absence d'installation, de mise à jour, de configuration, pour le logiciel de mail. On y retrouve également un argument phare du web 2.0 : la possibilité d'accéder à sa boîte mail depuis n'importe quel ordinateur connecté à Internet, n'importe quand, n'importe où.

10.5.2 Inconvénients

Côté inconvénients, il y a le fait qu'en cas d'absence de connexion, toute notre correspondance nous est inaccessible (à moins qu'on en ait sauvegardé tout ou partie sur un support à portée de main : clé USB, disque dur, *etc.*)

Le fait qu'il soit possible d'utiliser n'importe quel navigateur web pour accéder à notre boîte mail peut vite nous inciter à utiliser effectivement *n'importe quel* navigateur web,

et par là des ordinateurs en lesquels nous n'avons que très peu de raisons de placer notre confiance.

Ensuite, en fonction de la confiance que l'on place dans notre hébergeur d'email, il convient de se poser la question de la centralisation de nos données. L'usage massif du webmail nous amène à une situation où des milliers de boîtes mail, avec tout leur contenu, se retrouvent entre les mains des plus gros fournisseurs de service email, leur confiant ainsi la garde d'une montagne de données personnelles. Ces hébergeurs peuvent les utiliser à des fins commerciales, les livrer à diverses autorités, ou tout simplement les perdre. De plus, si l'on considère que notre correspondance est sensible d'une manière ou d'une autre, peut-être préfèrera-t-on ne pas la laisser reposer sur les épaules de personnes - car il y en a encore derrière les machines - qui n'ont pas particulièrement envie d'en porter la responsabilité. Tel fut probablement le cas courant août 2013 pour la société d'hébergement d'email Lavabit [2], qui hébergeait un compte email d'Edward Snowden et qui décida de stopper ses activités. Fermeture intervenue suite aux requêtes voire pressions de la part d'agences gouvernementales telles que la NSA ou le FBI.

Enfin, l'utilisation du webmail peut nous faire profiter pleinement d'un tas de publicités s'affichant dans notre navigateur web, lors de la consultation de notre boîte aux lettres informatique. Publicités qui peuvent d'ailleurs être choisies en fonction du contenu de nos mails.

page 33

10.6 Client mail

Un client mail ou client de messagerie est un logiciel qui sert à gérer ses emails : les recevoir, les lire, les envoyer, *etc.* Des clients mails connus sont par exemple *Outlook* de Microsoft ou *Thunderbird* de Mozilla. Il en existe de nombreux autres qui, malgré leurs différences, possèdent une interface globalement similaire, proche de celle des webmails.

Contrairement au webmail, où l'on va consulter ses emails stockés chez l'hébergeur en utilisant son navigateur web, ici la lecture des emails se fait grâce à un logiciel installé sur l'ordinateur. On se sert d'un périphérique de stockage local (disque dur de l'ordinateur utilisé, clé USB, *etc.*) comme espace de stockage des emails.

Pour reprendre notre petit schéma précédent, il faut remplacer les protocoles web par des protocoles email. Deux protocoles différents existent afin de recevoir son courrier, *IMAP* (pour *Internet Message Access Protocol*) et POP (pour *Post Office Protocol*).

Le premier, *IMAP*, permet de manipuler les emails stockés sur les serveurs d'email de notre hébergeur. À chaque connexion vers la boîte mail, une synchronisation a lieu afin d'avoir le même état (nombres d'emails, de brouillons, de dossiers, *etc.*) sur le serveur mail que sur notre client mail, et inversement. Et cela sans pour autant télécharger le contenu présent sur le serveur mail. Seuls les en-têtes des emails peuvent être rapatriés sur notre client mail par exemple.

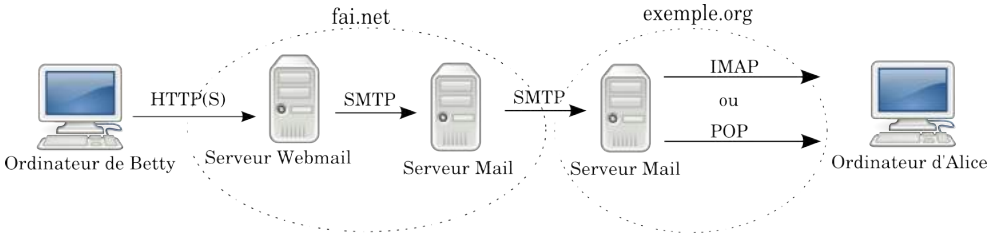
Le second protocole, *POP*, téléchargera les différents contenus de la boîte mail directement sur notre client mail, sans forcément en laisser de copie sur le serveur distant.

10.6.1 Avantages

Les avantages et inconvénients peuvent être spécifiques au protocole utilisé afin de recevoir son courrier, cela dit certains leur sont communs.

Tout d'abord, avec un client d'email, rien n'empêche d'avoir accès à sa boîte mail, dans le même état qu'au dernier relevé de courrier, même en l'absence de connexion à Internet. Il sera donc possible de lire, rédiger, supprimer des emails hors connexion,

2. Wikipédia, 2014, *Lavabit* [https://fr.wikipedia.org/wiki/Lavabit]



Betty utilise un webmail, Alice un client mail

sans bien sûr pouvoir ni les envoyer, ni en recevoir. De plus, l'usage d'un client mail nous évite d'avoir à subir la myriade de publicités dont le web est parsemé.

On n'aura également pas besoin de rajouter un certificat SSL à chaque fois que l'on se connecte à la page sécurisé de notre webmail si on utilise un système live amnesique comme *Tails*.

page 121

En utilisant le protocole *POP*, on profitera d'autres avantages comme la décentralisation des emails. Au lieu de laisser toute notre correspondance sur des serveurs distants, les courriers électroniques sont rapatriés sur l'ordinateur. Cela évite de laisser indéfiniment tous nos emails aux hébergeurs d'emails majeurs, mais aussi de dévorer trop d'espace disque chez les petits hébergeurs d'emails. Le fait que les emails finissent leur course sur le système du destinataire peut permettre également plus de prise sur leur gestion : concernant la suppression effective d'emails qui pourraient s'avérer être critiques par exemple. Enfin, on laisse moins de données à des entreprises qui n'ont que faire de la confidentialité de la correspondance. Attention cependant : nos emails transitent toujours par notre hébergeur, qui pourra y accéder et éventuellement en faire une copie avant qu'on ne les rapatrie.

10.6.2 Inconvénients

Pour utiliser un client mail, il va falloir configurer un logiciel de mail pour qu'il sache quelle boîte relever, à quel serveur se connecter et quel protocole utiliser.

Il est plus compliqué de consulter de cette manière ses emails depuis un ordinateur qui n'est pas le nôtre (chez des amis ou au travail par exemple), à moins d'utiliser le client mail d'un système *live* persistant (comme celui de *Tails*), installé sur une clé USB ou une carte SD.

De plus, dans le cas où le protocole POP est utilisé, le support de stockage sur lequel se trouve notre client mail sera le seul sur lequel sera stockée notre correspondance. En cas de perte du support sur lequel celle-ci est stockée (que ce soit le disque dur d'un ordinateur, la clé USB ou la carte SD sur lequel on a installé un système *live Tails* persistant), on peut dire adieu à ses précieux messages… sauf si on a pensé à en faire une sauvegarde.

tome 1 ch. 19

10.7 Échanger des emails en cachant son identité

L'objectif sera ici de cacher à un adversaire que nous sommes l'un des interlocuteurs d'un échange d'emails. Il s'agit peut-être d'un échange d'emails avec un dissident politique recherché ou bien avec une amie perdue de vue.

10.7.1 Définir une politique de sécurité

Notre souci principal va être de masquer les noms des personnes échangeant par email, ou du moins de rendre leur identification aussi difficile que possible.

Première étape : demander aux facteurs

Notre fournisseur d'emails est un nœud du réseau par lequel transitera obligatoirement notre correspondance numérique. Un adversaire s'y intéressant aurait donc de bonnes raisons de jeter un coup d'œil à cet endroit, d'autant plus que cela peut lui être très aisé.

De la même manière, les intermédiaires entre Betty et Alice, dont leurs FAI respectifs, n'auront qu'à regarder au bon endroit pour lire les en-têtes mail, qui peuvent livrer nombre d'informations, dont chez certains hébergeurs les adresses IP des correspondantes. Une telle attaque est plus que probable si le contenu des emails ou les protagonistes des échanges attirent l'attention d'autorités ayant des pouvoirs suffisants. Il est juste de se dire qu'en premier lieu, ne pas avoir une adresse email du genre `nom.prénom@exemple.org` est déjà un bon réflexe. Il va tout d'abord falloir penser à utiliser un pseudonyme, pour mettre sur pied une identité contextuelle.

page 30
page 53

Ceci dit, si « Kiwi Poilu » écrit régulièrement à Caroline Carot, Sofiane Carot et Francine Carot, un adversaire *pourrait* se dire qu'il appartient à la famille Carot, ou fait partie des intimes : les identités des gens à qui on écrit sont elles aussi révélatrices.

De plus, si l'on utilise un pseudonyme, mais qu'un adversaire observe que les emails qu'il surveille sortent de telle maison ou tel appartement, il peut effectuer le rapprochement. C'est pourquoi comme pour la navigation sur le web, l'utilisation du routage en oignon ou l'utilisation d'un système *live* amnésique prévu à cet effet permettent de brouiller des pistes remontant jusqu'à notre ordinateur.

page 69
tome 1 ch. 14

Enfin, le contenu des échanges peut permettre d'en apprendre suffisamment sur leurs auteurs pour mettre des noms dessus. Cacher une identité nécessite donc de faire attention non seulement aux en-têtes, mais aussi au contenu de l'email.

Pour protéger le contenu des emails des regards curieux, que ce soit pour lui-même ou pour ce qu'il peut divulguer sur les auteurs des mails, on utilisera le chiffrement d'emails.

page ci-contre

Deuxième étape : regarder sur l'ordinateur utilisé

Si le réseau Tor ainsi qu'un pseudonyme sont utilisés afin de protéger son identité, un attaquant potentiel peut essayer d'accéder aux traces laissées sur l'ordinateur. Afin de prouver que la personne qu'il suspecte est bien en possession du compte email en question.

tome 1 ch. 2

Pour se prémunir contre cette attaque, il est nécessaire de chiffrer son disque dur ou, mieux encore, d'éviter dès le départ de laisser ces traces en utilisant un système *live* amnésique.

tome 1 ch. 15
tome 1 ch. 14

C'est d'autant plus important si l'on utilise un client mail, car ce ne sont pas seulement des traces qui seraient laissées sur le système, mais également les mails eux-mêmes.

Troisième étape : attaquer Tor

Un attaquant capable de surveiller plusieurs points du réseau, par exemple la connexion utilisée *et* l'hébergeur d'email, pourrait être en mesure de défaire l'anonymat fourni par le réseau Tor.

Rappelons encore une fois que de nombreuses autres attaques sont envisageables contre le réseau Tor, et qu'il est impératif de bien saisir contre quoi il protège et contre quoi il ne protège pas.

page 70
page 72

10.7.2 Choisir parmi les outils disponibles

Plusieurs outils sont disponibles pour communiquer par email, le choix se fait donc en fonction des différents critères évoqués précédemment. On peut par exemple préférer

ne pas laisser ses emails sur le serveur de notre hébergeur, les lire et y répondre hors ligne ou au contraire ne pas vouloir télécharger de copie de ses emails et y accéder toujours en ligne.

10.7.3 Webmail

Le webmail étant un usage particulier du web, on se réfèrera au cas d'usage traitant de la navigation sur le web pour les questions relatives au *Tor Browser Bundle* ou à *Tails* – leurs avantages, leurs inconvénients, leur utilisation. Il faudra également prendre soin de vérifier les certificats ou autorités de certification qui offrent un chiffrement de la connexion vers le serveur de mail car un attaquant ayant les moyens de duper l'utilisateur à cet endroit-là sera en mesure de récupérer en clair tous les échanges avec le serveur de mail, dont login et mot de passe de la boîte mail.

page 79
page 121

De plus, si l'on utilise le webmail depuis *Tails* sur un ordinateur en lequel on peut avoir des doutes, notamment face à une attaque de type keylogger, on prendra soin d'utiliser un clavier virtuel lors de la saisie du mot de passe de son compte email.

tome 1 § 3.3
page 125

10.7.4 Client mail

Dans le cas où l'on préfère utiliser un client mail plutôt que faire du webmail, on peut au choix :

tome 1 ch. 14

- utiliser *Tails*, dans lequel est fourni le client Claws Mail. Les traces laissées localement seront alors effacées à l'extinction du système ;

page 127

- utiliser *Tails* et Claws Mail avec la persistance pour stocker le contenu de sa boîte sur une clé USB ou une carte SD. Des traces chiffrées seront alors stockées sur ce support ;

tome 1 § 14.5

- installer un client mail sur son système chiffré en utilisant l'outil installer un logiciel pour installer le paquet `claws-mail` ;

tome 1 ch. 15

- suivre l'outil utiliser Claws Mail. Des traces chiffrées seront alors laissées sur le disque dur de l'ordinateur.

tome 1 ch. 16
page 127

Mais de la même manière que pour un webmail, il faudra veiller à vérifier les certificats ou autorités de certification qui offrent un chiffrement de la connexion vers le serveur de mail.

page 121

10.8 Echanger des emails confidentiels (et authentifiés)

On souhaite ici cacher le contenu de nos emails, afin d'éviter qu'une autre personne que la destinataire ne puisse les lire, ce qui peut être utile lorsque le contenu de nos messages est *sensible* ou qu'il en dit long sur la personne l'ayant rédigé.

Pour définir notre politique de sécurité, il nous faut envisager l'utilisation du chiffrement selon plusieurs modalités.

10.8.1 Première étape : demander aux factrices

Sans mesure de protection particulière, les services d'hébergement d'email pourront lire le contenu des emails qui nous sont destinés. En effet, ce sont sur leurs serveurs que sont acheminés et stockés nos emails.

Connexion non-chiffrée aux serveurs mail

Nos messages peuvent être ainsi conservés pendant des années jusqu'à que nous les rapatrions ou les supprimions, voire plus longtemps encore si l'un des serveurs en fait une copie, dans le cadre d'une sauvegarde par exemple. D'où l'importance, notamment, de fermer des boîtes mail une fois leur raison d'être dépassée. Ceci a également l'avantage de ne pas occuper de l'espace disque pour rien chez l'hébergeur mail.

Il n'y a ici pas de grande différence entre l'usage de tel ou tel protocole, d'un client mail ou du webmail. L'usage du protocole POP avec un client mail bien configuré (téléchargement complet des emails, suppression sur le serveur distant), en admettant que l'on relève régulièrement notre courrier, évitera au mieux de laisser notre courrier prendre la poussière dans les serveurs de notre hébergeur.

La lecture de nos messages, qui relève de la violation du secret de la correspondance – tout comme lire une lettre qui ne nous serait pas destinée – ne demande aucun effort technique, pas même celui d'ouvrir une enveloppe. Elle est si simple que son utilisation a notamment été automatisée par Gmail, qui fait lire le contenu des emails de ses utilisateurs par des « robots » afin de détecter le spam [3], mais aussi afin de mieux cibler leurs utilisateurs pour, entre autres, leur « offrir » la publicité la plus adéquate.

> Ces « robots » ne sont ni des automates, ni des androïdes, mais de petits programmes qui parcourent « automatiquement » du contenu pour identifier quelque chose : par exemple, les « robots » de Google parcourent les pages web pour indexer les mots-clés pertinents qui pourraient être recherchés. De tels robots sont aussi utilisés par les flics pour leur signaler chaque fois qu'une personne utilise certains mots de leur supposé « dictionnaire des terroristes ».

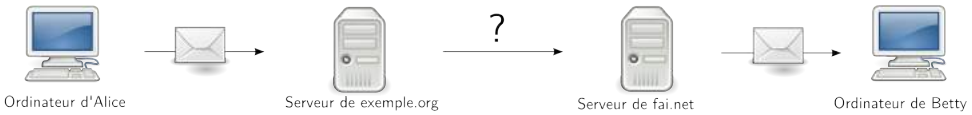
Concernant les intermédiaires situés entre les protagonistes de l'échange d'emails et les serveurs des hébergeurs d'emails respectifs, on peut avoir affaire à deux situations. La première, désormais plutôt rare, est celle où la connexion entre un protagoniste et son serveur mail n'est pas chiffrée. Dans ce cas-là, les différents intermédiaires verront passer entre leurs mains des cartes postales. Ils seront donc dans une situation similaire à celle des hébergeurs d'emails, à la différence près que les cartes postales ne feront que transiter… sauf s'il leur vient à l'idée de jeter un coup d'œil plus approfondi au courrier qu'ils transportent, que ce soit pour faire des statistiques afin d'améliorer la qualité de leur service, ou parce qu'on le leur demande gentiment.

La seconde situation est celle où la connexion entre un protagoniste et son serveur mail est chiffrée avec le protocole *TLS*. Ceci est possible quel que soit le protocole utilisé. Les intermédiaires entre un protagoniste et son serveur mail verront cette fois-ci des cartes postales mises dans des enveloppes. Enveloppes plus ou moins difficiles à ouvrir : en effet, si la connexion entre Alice et son serveur mail est effectivement chiffrée, Alice ne choisit pas de quelle manière elle l'est. De plus, l'hébergeur d'email ne sera pas affecté par le chiffrement et aura toujours accès à l'email dans son intégralité. Afin de profiter d'un chiffrement de la connexion efficace, il ne faudra pas faire une confiance *aveugle* à une autorité de certification, ou accepter un certificat sans le vérifier au préalable .

page 59
page 65
page 121

Enfin, rien n'assure que la connexion entre le serveur mail d'Alice et celui de Betty est chiffrée, auquel cas, le trajet de l'email se fera en partie à la manière d'une lettre, en partie à la manière d'une carte postale.

3. Le spam est une communication électronique non sollicitée, le plus souvent du courrier électronique.



Connexion chiffrée aux serveurs mail

Afin de s'assurer que le contenu de nos messages n'est lisible par aucun des intermédiaires, facteur compris, il nous est possible de les chiffrer directement sur notre ordinateur, avant même de les envoyer. Pour cela, on utilisera le standard de cryptographie asymétrique *OpenPGP*. Il serait également possible d'utiliser la cryptographie symétrique, mais ses limites nous amènent à le déconseiller fortement. page 59

En utilisant la cryptographie asymétrique, seule la personne destinataire, pour laquelle on aura effectué le chiffrement, sera en mesure de déchiffrer le message. N'oublions pas cependant que la cryptographie asymétrique possède également des limites qui peuvent permettre à un adversaire de révéler le contenu chiffré. page 68

En pratique, si ce n'est pas déjà fait, on commencera par importer la clé publique de notre destinataire. Il faudra alors vérifier son authenticité. De plus, si on compte établir une correspondance et donc recevoir des emails en retour il nous faudra également disposer d'une paire de clés : l'une sera utilisée par nos correspondants pour chiffrer des emails à notre intention, l'autre nous permettra de les déchiffrer. Si l'on n'a pas encore de paire de clés de chiffrement, suivre la recette pour en créer une et en avoir une bonne gestion. page 131 page 132 page 134

Après rédaction du message, on suivra les étapes nécessaires à son chiffrement. Ne reste plus qu'à l'envoyer ! page 138

Attention cependant, cette méthode permet de chiffrer le contenu de l'email et seulement le contenu. Elle ne modifiera en rien les en-têtes de l'email. page 30

10.8.2 Deuxième étape : regarder sur l'ordinateur utilisé

Supposons qu'un attaquant n'a pas accès aux données de nos hébergeurs, et ne peut pas écouter le réseau, mais qu'il peut venir se servir chez nous : quelles traces de nos échanges trouvera-t-il sur notre ordinateur ?

Si cet adversaire peut mettre la main sur notre ordinateur, ou sur celui de l'autre personne impliquée dans la communication, que ce soit en s'en emparant ou en arrivant à y installer un logiciel malveillant, il pourra avoir accès à tous les emails stockés sur celui-ci ainsi qu'aux traces laissées ; que ces traces soient dues au fonctionnement de la machine ou laissées par les soins de protagonistes. tome 1 ch. 3

Afin de se protéger d'un adversaire qui pourrait s'emparer de notre ordinateur, on prendra soin de chiffrer son disque dur pour lui compliquer l'accès aux données stockées sur celui-ci. Cela ne nous protègera pas contre un logiciel malveillant voulant exfiltrer ces données, d'où l'importance de n'installer que des logiciels de confiance. On pourra aussi utiliser un système *live* amnésique. Notons que si les emails stockés font partie d'un échange qui a été chiffré en utilisant la cryptographie asymétrique, quand bien même il a accès à l'ordinateur, l'adversaire ne pourra pas les lire, à moins qu'il ait accès à la clé secrète et connaisse la phrase de passe qui permet de l'utiliser. tome 1 ch. 15

10.8.3 Troisième étape : attaquer le chiffrement du support

Si l'on consulte ses emails sur une Debian chiffrée, les traces sur le disque dur de l'ordinateur seront chiffrées, que l'on utilise le webmail ou un client mail. Elles n'apprendront donc rien à un adversaire en l'état. Cependant certains adversaires pourraient avoir des moyens d'attaquer ce chiffrement. De plus, si la personne avec qui l'on converse par email ne fait pas de même, le niveau global de protection du contenu tome 1 § 5.1.4

sera nivelé par la plus faible des deux protections. En effet, avoir pris énormément de précautions et échanger des emails avec une personne ayant par exemple une Debian non chiffrée, ou allumée en permanence [4], peut être plus dangereux car l'on pourrait avoir une impression trompeuse de sécurité. D'autant plus s'il est aisé de localiser ou mettre un nom sur les protagonistes de l'échange.

En utilisant un logiciel de mail dans un système *live* amnésique, il n'y aura aucune trace sur l'ordinateur utilisé après extinction, mais il y en aura dans la partition persistante si on l'a configurée. Celles-ci seront chiffrées, ce qui revient au cas précédent d'une Debian chiffrée.

Pour ne pas laisser de traces sur l'ordinateur utilisé, chiffrées ou non, on pourra d'utiliser le système live *Tails* sans persistance et de profiter ainsi de son amnésie.

10.8.4 Quatrième étape : attaquer le chiffrement des messages

Un adversaire arrivant, d'une manière ou d'une autre, à mettre la main sur les emails chiffrés, peut essayer de s'attaquer au chiffrement du contenu afin de parvenir à le briser. Cet adversaire peut pour cela tirer parti des différentes limites du chiffrement.

tome 1 § 5.1.4

4. Lorsqu'elle est allumé, une machine dont le disque dur est chiffrée contient de nombreuses informations déchiffrées dans sa mémoire vive [tome 1 § 1.2.3].

Chapitre 11

Dialoguer

11.1 Contexte

Dans le cas d'usage précédent, on échangeait des messages de façon asynchrone, tout comme dans un échange épistolaire. Cependant on peut vouloir une communication synchrone, comme lors d'une communication téléphonique, que ce soit pour une réunion de travail sur un document sensible ou pour dialoguer avec une amie. Le plus simple pourrait être de se déplacer pour se rencontrer, ou de s'appeler - mais ce n'est pas toujours possible ou souhaitable. Cela peut également poser des problèmes. Parfois, la messagerie instantanée est une bonne alternative.

page 91

tome 1 ch. 9

Beaucoup de gens connaissent et utilisent régulièrement la messagerie de *Skype* (remplaçant de *MSN* ou *Windows Live Messenger* fournie par Microsoft) ou la messagerie interne de *Facebook*, pour ne citer que les exemples les plus connus. C'est pratique… mais il est possible de faire pratique sans renoncer à être discret.

11.2 Évaluer les risques

11.2.1 Que veut-on protéger ?

Les réponses possibles à cette question sont les mêmes que dans le cas de l'échange de messages. On peut vouloir protéger le contenu de l'échange, la localisation des protagonistes, leurs identités, leur lien, *etc.*

page 91

11.2.2 De qui veut-on se protéger ?

Ici aussi, les réponses sont proches de celles données dans le cas d'usage échanger des messages : on peut vouloir dissimuler tout ou partie de ces informations aux diverses machines par lesquelles elles transitent aussi bien qu'aux personnes qui pourraient y avoir accès.

page 91

Parmi lesdites machines, viennent tout d'abord les serveurs de messagerie instantanée utilisés par les différents correspondants.

page 29

Viennent ensuite les routeurs, situés sur le trajet entre les protagonistes de l'échange, notamment ceux de leurs FAI respectifs.

page 28

page 38

Enfin, des traces sont laissées sur les ordinateurs utilisés.

11.3 Définir une politique de sécurité

Posons-nous maintenant les questions exposées dans notre méthodologie en adoptant le point de vue de notre adversaire.

tome 1 ch. 7

11.3.1 Première étape : toutes les infos à disposition des curieux

La messagerie interne de Facebook, Skype, *etc.* permettent à beaucoup de gens de prendre connaissance d'informations qui ne les concernent pas : Facebook ou Microsoft verront passer l'intégralité de nos conversations sur leurs machines, et peuvent les archiver pour pouvoir y accéder ensuite. Les flics n'auront qu'à demander pour bénéficier des informations, et une faille de sécurité sur le serveur peut donner accès à de nombreux autres curieux. Sans oublier que Facebook change régulièrement ses réglages de confidentialité sans prévenir, et peut décider demain de rendre public ce qui est « privé » aujourd'hui.

Par ailleurs, Skype enregistre les conversations sur l'ordinateur utilisé, et donc n'importe quel voisin l'utilisant, cambrioleur l'emmenant ou amant jaloux peut accéder à cet historique aussi.

Mais Microsoft et Facebook n'ont pas inventé la messagerie instantanée et de multiples alternatives sont disponibles. Il existe de nombreux logiciels que l'on peut installer sur son ordinateur, qui permettront de communiquer selon divers protocoles : Skype, IRC, XMPP, *etc.*

Le fait d'utiliser un logiciel de confiance nous permettra de désactiver l'archivage des conversations, et donc de limiter les traces laissées sur notre ordinateur.

Il existe également des serveurs qui fournissent des adresses de messagerie instantanée et qui ne sont pas dans une position leur permettant de faire autant de recoupements que Google, Microsoft ou Facebook.

tome 1 ch. 15
tome 1 ch. 16
Pour suivre cette piste sur un système Debian (chiffré) installé précédemment, se référer à l'outil installer un logiciel pour installer `pidgin`. Si l'on utilise *Tails*, ce logiciel est déjà installé.

11.3.2 Deuxième étape : demander aux hébergeurs

En utilisant un client de messagerie instantanée et des serveurs variés, on ne centralise pas tous les liens et les dialogues entre les mêmes mains. Cependant, le contenu des conversations tout comme les parties qui communiquent restent accessibles à partir des ordinateurs par lesquels ils transitent.

S'il est souvent possible de paramétrer notre logiciel pour chiffrer la connexion jusqu'au serveur de messagerie, les dialogues restent accessibles au serveur. De plus, on ne peut en général pas garantir que le lien entre le serveur et l'autre correspondant soit aussi chiffré.

page 39
page 44
Un adversaire qui en a les moyens pourra donc s'adresser aux admins du serveur utilisé, voire aux organisations qui fournissent le réseau, pour obtenir des informations sur les conversations. Il pourra aussi tenter de « pirater » leurs machines.

La confidentialité des dialogues reste donc fortement liée à la confiance qu'on met dans les serveurs de messagerie que l'on utilise, voire dans les infrastructures du réseau et en particulier notre fournisseur d'accès.

page 59
Pour fortement compliquer la tâche d'un adversaire qui voudrait lire le contenu de nos dialogues, on pourra utiliser le chiffrement de bout en bout et disposer alors de **confidentialité**. Malheureusement, il n'y a pas actuellement d'implémentation du chiffrement de bout en bout qui permette des conversations de ce type en groupe. Cette solution sera donc limitée aux discussions à deux.

tome 1 ch. 15
tome 1 ch. 16
page 145
Pour suivre cette méthode sur un système Debian (chiffré) installé précédemment, suivre les outils installer un logiciel pour installer le paquet `pidgin-otr`, puis utiliser la messagerie instantanée avec OTR.

11.3.3 Troisième étape : les liens restent visibles

Si on utilise le chiffrement de bout en bout dans le cadre d'un dialogue en messagerie instantanée, un adversaire ne peut alors plus avoir accès au contenu de la conversation, à moins de casser le chiffrement utilisé, d'accéder à notre ordinateur, voire de le pirater.

page 68

page 47

Cependant, un adversaire qui a accès au réseau ou au serveur de messagerie utilisé peut toujours voir avec qui nous parlons. Pour cacher les liens, il faudra utiliser des identités contextuelles et se connecter de façon anonyme, par exemple en utilisant Tor. On a alors *confidentialité* grâce au chiffrement, mais aussi *pseudonymat*.

page 53

page 69

En utilisant un système *live* amnésique comme *Tails*, on s'occupe du même coup de la question des traces qui pourraient être laissées sur l'ordinateur utilisé. Sauf si on utilise la persistance, auquel cas des traces chiffrées seront conservées dans la partition persistante de la clé USB ou de la carte SD de *Tails*.

Pour suivre cette piste il nous faudra donc dans un premier temps, si l'on n'en a pas déjà, faire une clé USB, une carte SD ou un DVD *Tails*.

tome 1 ch. 14

Ensuite, après avoir démarré sur le support contenant *Tails*, il nous faudra définir une identité contextuelle à utiliser et mettre en place la persistance de *Tails* pour cette identité en activant l'option « Pidgin ».

tome 1 ch. 13

tome 1 § 14.5

Nous pourrons enfin suivre l'outil utiliser la messagerie instantanée avec OTR.

page 145

On combine ici deux critères : confidentialité et anonymat. À l'étape précédente, on a vu comment disposer de *confidentialité* avec le chiffrement OTR. Ici on vient de voir comment avoir *anonymat et confidentialité* en utilisant le chiffrement OTR sous *Tails* ainsi qu'une identité contextuelle. Cependant, on peut désirer l'*anonymat ou le pseudonymat seul*, c'est-à-dire sans confidentialité. En effet, on peut vouloir cacher qui on est sans cacher le contenu de nos conversations, par exemple pour discuter dans des « salons » publics traitant de pratiques sexuelles considérées comme déviantes. Pour suivre cette piste on démarrera alors *Tails* puis on utilisera *Pidgin* avec un compte créé automatiquement pour l'occasion [1] *sans* utiliser le chiffrement OTR.

tome 1 § 14.4

page 145

11.4 Les limites

Tout d'abord, cette méthode reste vulnérable aux éventuelles attaques sur le chiffrement, dont on vient de parler et aux attaques sur Tor.

page 81

Mais il existe aussi quelques limites spécifiques aux conversations en temps réel. Ainsi, l'état « en ligne » ou « hors ligne » d'une identité est en général accessible publiquement. Un adversaire peut ainsi voir quand une identité est connectée, et éventuellement corréler plusieurs identités : parce qu'elles sont toujours en ligne en même temps ; ou au contraire parce qu'elles ne sont jamais en ligne en même temps mais souvent successivement, *etc.*

> Pour que des identités apparaissent comme étant « toujours en ligne », il est possible d'utiliser un « ghost » ou proxy [2] sur un ordinateur en qui l'on a confiance, qui est toujours allumé et connecté au serveur de messagerie instantanée. C'est ainsi cet ordinateur, et non pas le serveur, qui « voit » quand on est connecté ou pas, et cet état n'est plus public. La mise en place d'une telle infrastructure dépasse toutefois pour l'instant les ambitions de ce guide.

1. Lors du démarrage de *Tails*, deux comptes de messagerie instantanée sont générés automatiquement [https://tails.boum.org/doc/anonymous_internet/pidgin/index.fr.html]

Ensuite, dans le cas particulier où l'anonymat (ou le pseudonymat) est prioritaire sur d'autres contraintes, par exemple si l'on souhaite discuter dans un salon public, d'autres limites s'ajoutent à celles évoquées ci-dessus. Ainsi, une identité contextuelle risque toujours de finir reliée à une identité civile, comme nous l'avons vu dans la partie sur les pseudonymes. En effet, même sous un pseudonyme, le fond et la forme de nos conversations peuvent en dire très long sur la personne se trouvant derrière le clavier.

page 54

Il est bon de garder en mémoire le fait que quand on essaye de définir une politique de sécurité lors d'une relation entre plusieurs personnes, que ce soit au téléphone, dans le cas d'échanges d'emails ou encore ici pour la messagerie instantanée, le niveau global de sécurité sera nivelé par le niveau de sécurité du protagoniste le moins précautionneux. En effet, si l'on prend par exemple soin d'utiliser *Tails* afin de ne laisser aucune trace de notre conversation sur l'ordinateur, alors que notre interlocuteur utilise son système d'exploitation habituel sans protection particulière, alors ce dernier sera sans doute le point le plus faible de la politique de sécurité de notre communication.

page 91

Enfin, comme cela a déjà été dit, le chiffrement OTR ne permet pas à l'heure actuelle de converser à plus de deux à la fois. Des recherches avancent cependant dans ce sens [3].

En attendant, et à condition d'aimer bidouiller, il est d'ores et déjà possible de mettre en place son propre serveur de messagerie instantanée (par exemple XMPP) sur un service caché Tor [page 71].

2. Wikipédia, 2014, *Proxy* [https://fr.wikipedia.org/wiki/Proxy].
3. Ian Goldberg, et Al., 2009 *Multi-party Off-the-Record Messaging*, CACR Tech Report 2009-27 [http://www.cacr.math.uwaterloo.ca/techreports/2009/cacr2009-27.pdf] (en anglais) ; Jacob Appelbaum, et Al., 2013, *mpOTR* [https://github.com/cryptocat/mpOTR] (en anglais).

Troisième partie

Outils

Dans cette troisième partie, nous expliquerons comment appliquer concrètement quelques-unes des pistes évoquées précédemment.

Cette partie n'est qu'une annexe technique aux précédentes : une fois comprises les problématiques liées à l'intimité dans le monde numérique ; une fois les réponses adaptées choisies, reste la question du « Comment faire ? », à laquelle cette annexe apporte certaines réponses.

Du bon usage des recettes

Les outils et recettes qui suivent sont des solutions extrêmement partielles, qui ne sont d'aucune utilité tant qu'elles ne font pas partie d'un ensemble de pratiques articulées de façon cohérente.

Piocher dans cette boîte à outils sans avoir, au préalable, étudié la partie sur le choix d'une réponse adaptée et défini une *politique de sécurité*, est un moyen remarquable de se tirer une balle dans le pied en croyant, à tort, avoir résolu tel ou tel problème.

page 77
tome 1 ch. 7

On ne peut pas faire plaisir à tout le monde

Pour la plupart des recettes présentées dans ce guide, nous partons du principe que l'on utilise GNU/Linux avec le bureau GNOME ; ces recettes ont été écrites et testées sous Debian 7.0 (surnommée Wheezy) [1] et *Tails* [2] (*The Amnesic Incognito Live System*).

Pour autant, celles-ci sont généralement adaptables à d'autres distributions basées sur Debian, telles qu'Ubuntu [3] ou gNewSense [4].

Si l'on n'utilise pas encore GNU/Linux, on pourra consulter les cas d'usage du premier tome, au chapitre un nouveau départ ou utiliser un système *live*.

tome 1 ch. 8
tome 1 ch. 14

1. http://www.debian.org/releases/wheezy/index.fr.html
2. https://tails.boum.org/index.fr.html
3. http://www.ubuntu-fr.org/
4. http://www.gnewsense.org/Main/HomePage.fr

De la bonne interprétation des recettes

Avant de passer aux recettes elles-mêmes, quelques remarques transversales nous ont paru nécessaires.

Les procédures sont présentées pas à pas et expliquent, chaque fois que c'est possible, le sens des actions que l'on propose d'effectuer. Une utilisation efficace de ces outils nécessite de s'entendre sur quelques points :

- L'ordre dans lequel chaque recette est développée est d'une importance capitale. Sauf mention contraire, il est simplement inimaginable de sauter une étape pour ensuite revenir en arrière : le résultat, si jamais ces opérations désordonnées en donnaient un, pourrait être soit différent de celui escompté, soit tout bonnement catastrophique.
- Dans le même ordre d'idée, les actions indiquées doivent être effectuées à la lettre. Omettre une option ou ouvrir le mauvais dossier peut avoir pour effet de totalement modifier le résultat escompté.
- De manière générale, la bonne compréhension de ces recettes demande d'y accorder un minimum d'attention et de vivacité d'esprit. On ne peut pas tout réexpliquer à chaque fois : il est implicite d'avoir auparavant consulté et compris les explications des « cas d'usage », dont ces recettes ne sont que la dernière étape.

page 77

- Enfin, les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de ce guide, qui est disponible sur le site web https://guide.boum.org/.

Chapitre 12

Installer et configurer le Tor Browser Bundle

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : une demi-heure à une heure.

Nous avons vu que, lors de nos navigations sur le web, les sites visités peuvent enregistrer notre adresse IP et donc qu'un adversaire peut facilement remonter à nous par ce biais. D'où, parfois, la nécessité de dissimuler cette adresse IP. Tor est un logiciel permettant de faire transiter notre connexion au sein d'un réseau de « nœuds », masquant ainsi notre IP réelle. C'est le routage en oignon.

page 12

page 69

Pour pouvoir utiliser le réseau d'anonymisation Tor, il faut paramétrer le logiciel Tor lui-même, mais également les logiciels qui vont l'utiliser, comme le navigateur web par exemple. Ces paramétrages sont souvent complexes, à tel point qu'il est difficile d'être sûr de l'anonymat qui en résulte.

C'est pourquoi il est conseillé, pour utiliser Tor, de se servir soit d'un système live dédié à cet usage, soit d'utiliser un « kit prêt à l'emploi » : le *Tor Browser Bundle* (*TBB*). C'est un outil qui permet d'installer et d'utiliser très facilement Tor sur un système « classique ». Aucun paramétrage ne sera nécessaire et tous les logiciels indispensables à une navigation sous Tor y sont inclus.

tome 1 ch. 14

Le Tor Browser Bundle rassemble :

- le navigateur web Firefox, paramétré pour utiliser Tor ;
- le logiciel Tor ;
- un lanceur, pour démarrer le tout en un simple double-clic.

Attention : il faut bien garder à l'esprit que le *Tor Browser Bundle* ne procure *pas* un anonymat pour l'ensemble de l'ordinateur : seules les connexions vers les sites web à l'aide du navigateur inclus dans le *TBB* passent par Tor. **Toutes les autres connexions (client mail, agrégateurs de flux RSS, autres navigateurs, *etc.*) ne sont pas anonymisées**. De plus, les traces de navigation, comme les cookies ou les mots de passe, seront probablement enregistrées sur le disque dur, de même que les documents téléchargés. Enfin, il arrive parfois, que lors de la navigation, on clique sur un lien qui lance automatiquement un logiciel (lecteur de musique par exemple) qui lui ne passera *pas* par Tor. Des indices sur la nature de notre navigation pourraient alors fuiter.

On expliquera ici comment installer le *Tor Browser Bundle* sur une Debian chiffrée.

tome 1 ch. 15

Pour utiliser un système se connectant à Internet uniquement *via* Tor et pouvoir utiliser Tor avec d'autres logiciels qu'un navigateur, il faudra se tourner vers un système live comme *Tails*.

tome 1 ch. 14

12.1 Télécharger et vérifier le *Tor Browser Bundle*

12.1.1 Vérifier l'architecture de son système d'exploitation

tome 1 § 1.2.2

Avant d'installer le *Tor Browser Bundle*, il faut déterminer l'architecture de notre système d'exploitation (Debian) ; savoir si elle est en 32 ou 64 bits. Pour ce faire, cliquer sur votre nom d'utilisateur (en haut à droite), puis sur *Paramètres système*. Une fenêtre s'ouvre avec plusieurs icônes. Cliquer sur *Détails*, tout en bas à droite. Nous avons à présent différents renseignements sur notre machine. En face de *Type d'OS* il y a soit *32 bits*, soit *64 bits*. Noter cette information dans un coin et fermer la fenêtre.

12.1.2 Télécharger le programme

Nous pouvons maintenant télécharger le *Tor Browser Bundle*. Aller sur `https://www.torproject.org` et cliquer sur le gros bouton *Download Tor*. Puis, dans la colonne de droite listant différents types de systèmes d'exploitation, cliquer sur *Linux/Unix*. Là, dans l'encadré correspondant à votre architecture (c'est ce que l'on a noté à l'étape précédente) sélectionner la langue désirée, par exemple *Français*, et enfin cliquer sur *Download*.

Une fenêtre proposant d'ouvrir le fichier s'affiche. Choisir *Enregistrer le fichier* et cliquer sur *OK*. Une fenêtre de téléchargement s'ouvre alors. Quand le téléchargement est terminé, on peut fermer celle-ci. Nous sommes à présent de nouveau sur le site du projet Tor.

12.1.3 Télécharger la signature du programme

page 62

Nous allons à présent télécharger la signature du fichier pour pouvoir l'authentifier.

page 63

Pour ce faire, il faut faire un clic-droit sur *(sig)* situé juste au-dessous du bouton *Download Tor Browser Bundle* et à gauche du menu déroulant de sélection de la langue. Cliquer sur *Enregistrer la cible du lien sous…* une fenêtre nous demande de choisir où enregistrer ce nouveau fichier. Il faut l'enregistrer dans le même dossier que le fichier précédent (par défaut dans *Téléchargements*).

Une fois la signature téléchargée, on peut fermer la fenêtre des téléchargements et la fenêtre du site du projet Tor.

12.1.4 Vérifier l'authenticité du programme

page 131

Importer la clé OpenPGP qui signe le Tor Browser Bundle :

```
Erinn Clark <erinn@torproject.org> 0x63FEE659
```

Il faut être attentif qu'il s'agisse de la bonne clé, étant donné que des petits plaisantins pas forcément bien intentionnés s'amusent à publier des clés GnuPG pour cette identité. Le projet Tor maintient une page web [`https://www.torproject.org/docs/signing-keys.html.en`] (en anglais) listant les véritables clés à utiliser. C'est un bon point de départ pour ne pas se tromper.

L'empreinte de cette clé observée par les auteurs de ce guide est, en admettant que c'est un exemplaire original que l'on a entre les mains :

```
8738 A680 B84B 3031 A630 F2DB 416F 0610 63FE E659
```

C'est un premier pas pour utiliser l'outil de vérification de l'authenticité d'une clé, qui est nécessaire pour s'assurer de ne pas avoir affaire à une usurpation d'identité de la part de quelqu'un qui voudrait nous faire utiliser un faux TBB. page 132

Une fois rassuré, suivre l'outil vérifier la signature numérique, afin de vérifier l'authenticité du TBB qui a été téléchargé. Si elle est confirmée, on peut passer à la suite. Dans le cas contraire, il peut être nécessaire de re-télécharger le TBB et sa signature et de recommencer le processus de vérification. page 139

12.2 Décompresser le Tor Browser Bundle

Tout d'abord, copier le fichier à installer dans son dossier personnel : pour ce faire, se rendre dans le dossier où l'on a téléchargé le *Tor Browser Bundle*. Effectuer un clic-droit sur le fichier *tor-browser-gnu-linux-….tar.xz* et sélectionner *Couper* dans le menu contextuel qui apparaît. Se rendre alors dans son dossier personnel, effectuer un clic-droit et sélectionner *Coller*.

Pour terminer d'installer le *Tor Browser Bundle*, nous allons le décompresser :

Dans le dossier personnel, faire un clic-droit sur le fichier *tor-browser-gnu-linux-….tar.xz* et cliquer sur *Extraire ici*. Un dossier s'appellant *tor-browser_fr* est créé. On peut ensuite supprimer le fichier *tor-browser-gnu-linux-….tar.xz* : le TBB est installé !

12.3 Lancer le Tor Browser Bundle

Pour lancer le TBB, il faut se rendre dans le *Dossier personnel*, puis dans le dossier *tor-browser_fr*. Là il faut double-cliquer sur *start-tor-browser*, une fenêtre s'ouvre nous demandant si on veut lancer *start-tor-browser* et comme c'est ce que l'on veut, on clique sur *Lancer* à droite. Le lanceur du TBB s'ouvre et nous propose soit de *Se connecter* au réseau Tor directement, soit de le *Configurer* avant notre première connexion au réseau Tor *via* le TBB. Si tout se passe bien, après avoir cliqué sur *Se connecter*, en quelques secondes, un navigateur web s'ouvre, dans lequel il est inscrit « Félicitations ! Ce navigateur est configuré pour utiliser Tor ». L'adresse IP du nœud de sortie s'affiche si l'on clique sur « Tester les paramètres du réseau Tor ».

Il peut arriver dans certaines conditions, comme dans le cas du bloquage des connexions vers le réseau Tor par des fournisseur d'accès à Internet, que le TBB n'arrive pas à se connecter. Dans ce cas, ou simplement pour plus d'informations, consulter la documentation disponible sur le site web du projet Tor [1].

1. Tiré et adapté de Torproject, *Le Guide Rapide Utilisateur* [http://www.torproject.org.in/dist/manual/short-user-manual_fr.xhtml]

Chapitre 13

Naviguer sur le web avec Tor

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 5 à 10 minutes.

L'objectif de cet outil est de naviguer sur le web de façon anonyme en utilisant Tor. Il n'y a pas beaucoup de différence avec l'utilisation d'un navigateur web « classique » qu'on considèrera comme un prérequis. La documentation de *Tails* sur *NetworkManager*[1] vous expliquera comment connecter votre ordinateur à Internet.

page 69

13.1 Lancer le navigateur

Si vous utilisez le système live *Tails*, le logiciel Tor est automatiquement démarré une fois la connexion Internet établie, et il suffit alors de lancer le navigateur web. Sinon, il vous faudra utiliser le *Tor Browser Bundle* après l'avoir installé.

tome 1 ch. 14

page 111

Attention, lorsqu'on utilise le *Tor Browser Bundle*, seul le navigateur lancé par *start-tor-browser* fournit l'anonymat procuré par Tor.

13.2 Quelques remarques sur la navigation

Une fois le navigateur lancé, on peut s'en servir presque comme d'un navigateur ordinaire. Cependant, certains détails sont à noter.

Tout d'abord, il faut avoir bien compris contre quoi Tor protège, mais surtout contre quoi il ne protège pas afin de ne pas faire n'importe quoi en se croyant protégé.

page 72

Les sites web consultés peuvent savoir que l'on se connecte *via* le réseau Tor. Certains, comme Wikipédia, utilisent cela pour bloquer l'édition anonyme. D'autres, comme Google, demanderont de résoudre des défis appelés « captcha »[2] pour montrer qu'on est bien un humain avant d'accéder à leurs services.

Enfin, certaines fonctionnalités sont désactivées pour protéger l'anonymat. C'est notamment le cas de Flash, technologie utilisée par beaucoup de sites proposant des vidéos en streaming.

page 26

1. https://tails.boum.org/doc/anonymous_internet/networkmanager/index.fr.html
2. Wikipédia, 2014, *Captcha*[3]

Chapitre 14

Choisir un hébergement web

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : une demi-heure à une heure.

L'objectif de cette recette est de trouver où héberger un document sur le web. Les possibilités sont trop nombreuses pour pouvoir apporter une réponse « clé en main » à cette question. De plus, conseiller une petite liste d'hébergeurs, sur lesquels seraient centralisés beaucoup de contenus « à risque », ne semble pas une bonne idée. Cette recette donnera donc plutôt quelques pistes pour bien choisir un hébergement.

14.1 Quelques critères de choix

Il existe de très nombreux hébergeurs, à tel point que l'on peut vite se sentir perdus dans la jungle des possibilités. Voici quelques critères pour se poser les bonnes questions. On parlera ci-dessous de document, mais ces critères valent aussi pour un projet plus ambitieux, tel qu'un blog ou un documentaire vidéo.

- **Type d'organisation** : beaucoup de sites proposent d'héberger des documents « gratuitement ». Nombre d'entre eux sont des services commerciaux qui trouvent un intérêt à publier du contenu créé par leurs utilisateurs. Mais il existe aussi des associations ou des collectifs qui hébergent également des projets, sous certaines conditions ; page 50
- **Conditions d'hébergement :** si le document ne plaît pas à l'hébergeur, rien ne l'empêche de le supprimer sans même nous avertir. Sa charte (que l'on doit accepter lors de l'hébergement de notre document) peut souvent nous donner une idée de ce que l'hébergeur tolère ou non ;
- **Résistance aux pressions :** l'État peut lui aussi vouloir empêcher que notre document reste en ligne. Il lui suffit, dans bien des cas, d'intimider l'hébergeur pour que ce dernier supprime notre document. En effet, selon l'hébergeur choisi, celui-ci peut supporter plus ou moins la pression : certains attendront qu'un recours à la justice soit effectué, tandis que d'autres supprimeront notre document dès le premier email un peu menaçant ;
- **Suppression du document :** inversement, on peut vouloir à un moment supprimer notre document. Or, l'hébergement de documents étant un service que l'on remet dans les mains d'autres personnes plus ou moins de confiance, on ne peut pas avoir la garantie que nos fichiers seront réellement effacés. Mieux connaître l'hébergeur peut dans certains cas nous donner plus de garanties.
- **Risques pour l'hébergeur :** selon le contenu de notre document, il peut faire courir un risque à l'hébergeur, en particulier s'il s'agit d'un hébergeur qui ne souhaite pas collaborer avec les flics. Il est alors nécessaire de se demander si l'on est

prêt à faire courir un risque à un hébergeur, qui peut être amené à disparaître en cas de répression.
- **Taille du document :** si notre document est « trop gros », certains hébergeurs refuseront de le prendre. Cela peut également être le cas si notre document est « trop petit ». La taille autorisée est spécifiée dans certaines offres mais attention : certains hébergeurs rendent payantes des fonctionnalités comme l'hébergement de très gros fichiers.
- **Durée d'hébergement :** selon les hébergeurs, de nombreuses offres existent quant à la durée de l'hébergement. Par exemple, certains suppriment automatiquement le document au bout d'un délai, d'autres s'il n'a pas été téléchargé pendant un certain temps, *etc.*

14.2 Type de contenu

Maintenant qu'on a quelques critères de choix en tête, essayons de rendre cela plus concret. L'hébergement adapté à notre projet dépend tout d'abord du type de contenu que l'on souhaite publier : texte, image, vidéo, son, *etc.*

14.2.1 Publier du texte

Publier du texte est souvent ce qu'il y a de plus simple.

Si le texte à publier est en rapport avec un autre texte déjà publié, il est souvent possible de poster un commentaire, que ce soit sur un blog, un forum ou autre site participatif. Pour ce genre de publication, l'inscription n'est pas forcément obligatoire, mais cela ne veut en aucun cas dire que la publication est anonyme si l'on ne prend page 69 pas de précautions particulières, comme par exemple utiliser le routage en oignon. De plus, notre texte étant un commentaire et non un sujet principal, il n'est pas forcément mis en avant sur le site.

Il est aussi possible de publier un texte sur un site ou un blog existant. Il faudra alors l'envoyer au site en question *via* un formulaire ou par email et la publication dépendra alors du ou des admins. Certains sites [1] proposent la publication libre d'articles sur un thème donné.

14.2.2 Publier un blog ou autre site

Si l'on souhaite publier régulièrement des textes, on peut alors choisir d'administrer un blog : de nombreuses organisations proposent des blogs déjà configurés et faciles à utiliser. On pourrait également administrer un site web, cette méthode demande toutefois un peu d'apprentissage.

Dans de nombreuses villes, des groupes de personnes s'intéressant au logiciel libre ou à la liberté d'expression sur Internet peuvent être de bon conseil. Quelques listes sont aussi disponibles sur le web :

- une liste d'hébergeurs de blogs sur l'annuaire ouvert dmoz [http://www.dmoz.org/World/Français/Informatique/Internet/Weblogs/Outils/Hébergement/]
- une liste de grosses plateformes de blogs sur Wikipédia [https://fr.wikipedia.org/wiki/Catégorie:Hébergeur_de_blogs]
- une autre liste de plateformes de blogs [http://www.allists.com/2033]
- une liste de services web libres sur le wiki de la communauté francophone d'Ubuntu [http://doc.ubuntu-fr.org/liste_de_services_web_libres]
- une liste d'hébergeurs sur le wiki des hébergeurs libres [http://www.hebergeurslibres.net/wakka.php?wiki=ListeHebergeurs]
- Il existe aussi l'hébergeur noblogs.org [http://noblogs.org].

1. Par exemple les sites du réseau Indymedia, qui proposent « d'assurer à tous la liberté de créer et de diffuser de l'information » Wikipédia, 2014, *Indymedia* [https://fr.wikipedia.org/wiki/Indymedia]

14.2.3 Publier des images

On veut souvent pouvoir publier des images, pour accompagner son texte. La plupart des sites où l'on peut publier du texte proposent d'en inclure dans notre article. Ils nous proposent alors soit de prendre des images depuis notre ordinateur (ils hébergent alors ces dernières sur leur serveur) soit d'indiquer l'adresse d'images déjà hébergées sur un autre serveur.

Il existe aussi des sites spécifiques de partage de photos :

- une liste de sites d'hébergement d'images sur l'annuaire ouvert dmoz [http://www.dmoz.org/World/Français/Informatique/Images/Web/Hébergement/]
- une liste de sites de partage de photos [http://www.allists.com/2042]

14.2.4 Publier une vidéo ou un son

Pour publier un contenu sonore ou une vidéo, des sites spécialisés existent. Ils permettent aux internautes qui les visitent de « lire en ligne » le son ou la vidéo, sans avoir à télécharger le fichier. Quelques listes de sites de ce type :

- [la page Wikipédia sur les sites de partage de fichiers](https ://fr.wikipedia.org/wiki/Site_d'hébergement_de_fichiers#H.C3.A9bergement_de_vid.C3.A9
- une liste de sites de partage de vidéos sur un blog [http://www.blog.niums.com/2007/net/quoi-de-neuf-sur-le-web/partage-de-videos-en-ligne-que-choisir/]

14.2.5 Publier un gros fichier téléchargeable

Pour publier des documents que l'on souhaite téléchargeables on va aller voir du côté des services de téléchargement direct de fichiers (ou *DDL* pour *Direct Downloading Link*). En français cela signifie « lien de téléchargement direct » : on « poste » notre fichier sur un serveur de téléchargement direct. On obtient alors un lien (une adresse web) qui lorsqu'on la tape dans notre navigateur, nous permet de lancer le téléchargement du fichier. Quelques listes de tels services :

- liste de 10 sites de partage de fichier sur clubic [http://www.clubic.com/article-87134-2-solutions-partager-gros-fichiers.html]
- liste de sites de partage de fichier sur Wikibooks [https://fr.wikibooks.org/wiki/Partage_de_fichiers_sur_Internet/Les_sites_d%27h%C3%A9bergement_de_fichiers_en_un_clic/Partager_un_fichier_en_un_clic]

14.3 En pratique

De façon plus concrète, il faut en premier lieu choisir l'hébergeur du fichier. Les critères présentés auparavant aident à effectuer ce choix. Il est très important de choisir un hébergeur en toute connaissance de cause car notre anonymat peut dépendre en partie de ce choix.

Il faut maintenant héberger le fichier à proprement parler. La méthode *exacte* est différente selon l'hébergeur, mais le principe reste le même. On va tout d'abord ouvrir notre navigateur web et l'utiliser de façon discrète. Ensuite on va se rendre sur le site de l'hébergeur et trouver la page où « déposer » notre fichier (*upload* en anglais). Là il faudra suivre la méthode spécifique à l'hébergeur afin de lui transmettre notre fichier. En général, cette méthode est facile à suivre et, même si elle varie, relativement similaire d'un hébergeur à l'autre. Une fois l'*upload* terminé, l'adresse web à laquelle se trouve le fichier est affichée.

page 79

Il est parfois nécessaire d'entrer une adresse email afin de recevoir ce lien web : le cas d'usage sur les échanges par email et le chapitre sur les identités contextuelles vont nous permettre de décider quelle adresse email fournir le cas échéant.

page 95
page 53

Une fois le lien obtenu, on peut le diffuser de la manière qui nous convient le mieux. Les personnes qui disposeront du lien pourront télécharger le fichier en le saisissant dans la barre d'adresse d'un navigateur web.

Chapitre 15

Ajouter un certificat électronique à son navigateur

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 15 à 30 minutes.

Nous avons vu dans la première partie de ce guide, qu'afin d'établir une connexion chiffrée il fallait souvent faire confiance à une autorité de certification (AC). La plupart du temps, les AC sont déjà enregistrées, dans le navigateur web par exemple. Mais ce n'est pas toujours le cas : dans cette situation, votre navigateur ou autre logiciel vous présentera un message vous expliquant qu'il n'a pas pu authentifier le certificat du site.

page 65

Il arrive également que le site visité n'utilise pas les services d'une autorité de certification, par manque de confiance, ou de moyen financier. Il faut alors vérifier son certificat.

15.1 Vérifier un certificat ou une autorité de certification

Que ce soit pour le certificat d'une AC, ou pour celui d'un site en particulier, il est nécessaire de le vérifier avant de l'installer. Sans cela, la connexion sera bien chiffrée, mais pas *authentifiée*. Autrement dit, on chiffera bien la communication, mais sans savoir vraiment avec qui – autant dire qu'à part se donner une fausse impression de sécurité, cela ne sert pas à grand-chose.

page 64

Vérifier un certificat signifie la plupart du temps visualiser son empreinte numérique et la comparer avec une autre source afin de s'assurer que celle-ci est correcte. Utilisez de préférence l'empreinte numérique de type *SHA1*, et non celle de type *MD5*, cette dernière n'étant plus considérée comme sûre [1].

tome 1 § 5.2

Pour l'autorité de certification CaCert par exemple, on obtiendra une chaîne de caractères de ce genre :

```
13:5C:EC:36:F4:9C:B8:E9:3B:1A:B2:70:CD:80:88:46:76:CE:8F:33
```

Encore reste-t-il à trouver d'autres sources permettant d'obtenir cette empreinte. Il existe quelques techniques pour essayer de s'assurer de l'authenticité d'un certificat :

1. Chad R Dougherty, 2008, *MD5 vulnerable to collision attacks* [http://www.kb.cert.org/vuls/id/836068] (en anglais).

- si une personne de confiance à proximité de vous utilise déjà le site ou l'AC en question et a déjà vérifié son certificat, vous pouvez comparer l'empreinte du certificat qu'elle connaît avec celle qui vous est présentée. Vous pouvez également la demander par email à des personnes qui vous l'enverront de façon chiffrée *et signée* pour plus de sécurité. C'est encore mieux si vous avez à votre disposition plusieurs de ces personnes, qui auraient vérifié ce certificat en utilisant chacune différentes connexions à Internet. Il faut alors suivre la démarche expliquée plus bas pour retrouver l'empreinte d'un certificat déjà installé dans le navigateur de cette personne.
- si vous avez accès à plusieurs connexions à Internet depuis l'endroit où vous êtes, par exemple en zone urbaine où l'on trouve beaucoup d'accès Wi-Fi, visitez le site ou téléchargez le certificat de l'AC en utilisant plusieurs d'entre elles et comparez l'empreinte du certificat qui vous sera présentée à chaque fois.
- si vous utilisez Tor, vous pouvez profiter du changement de circuit, et donc de nœud de sortie sur Internet, pour vérifier à plusieurs reprises l'empreinte du certificat. Cela évitera qu'un adversaire ayant la main sur le nœud de sortie ou étant placé entre le nœud de sortie et le site consulté puisse usurper son identité.

page 70
page 64

Pour savoir si votre nœud de sortie a changé, visitez en utilisant votre navigateur Tor un site comme celui de torproject [https://check.torproject.org/] qui vous indique l'IP de votre nœud de sortie. À chaque fois que celle-ci change, visitez le site souhaité ou téléchargez le certificat de l'AC, et comparez son empreinte avec celle collectée les fois précédentes. Ne les acceptez pas encore. Au bout de quelques essais réussis, la crédibilité qu'il s'agisse du bon certificat devient suffisamment grande pour l'accepter. Enfin, c'est à vous d'en juger en fonction de votre politique de sécurité !

tome 1 ch. 7

Ces techniques utilisées isolément ne sont pas forcément très robustes, mais leur utilisation conjointe procurera une crédibilité suffisante dans le fait que le certificat que vous allez utiliser est le bon. Et que personne n'aura réussi à vous tromper.

15.2 Ajouter un certificat

Il est possible d'ajouter un certificat d'un site particulier plutôt qu'une autorité de certification, notamment dans les cas où le certificat est auto-signé : c'est-à-dire lorsque celui-ci n'est pas émis par une autorité de certification mais par l'hébergeur lui-même. De la même manière, cette manipulation est parfois nécessaire lorsqu'il s'agit d'ajouter le certificat d'un serveur mail dans un client mail. On peut également préférer cette méthode, permettant de décider au cas par cas, plutôt que de placer sa confiance dans des autorités de certification : on a vu qu'elles n'en étaient pas toujours dignes.

page 65

Dans le cas d'un site web, après avoir rentré une adresse commençant par *https*, il arrive qu'on obtienne un message d'avertissement comme suit :

> **Cette connexion n'est pas certifiée**
>
> Vous avez demandé à Iceweasel de se connecter de manière sécurisée à **site.com**, mais nous ne pouvons pas confirmer que votre connexion est sécurisée.
>
> Normalement, lorsque vous essayez de vous connecter de manière sécurisée, les sites présentent une identification certifiée pour prouver que vous vous trouvez à la bonne adresse. Cependant, l'identité de ce site ne peut pas être vérifiée.
>
> **Que dois-je faire ?**
>
> Si vous vous connectez habituellement à ce site sans problème, cette erreur peut signifier que quelqu'un essaie d'usurper l'identité de ce site et vous ne devriez pas continuer.
>
> *Sortir d'ici !*

► **Détails techniques**

► **Je comprend les risques**

La notion d'usurpation d'identité évoquée dans le message précédent a été abordée dans un chapitre de la première partie. Une fois cet avertissement parcouru, on peut cliquer sur *Je comprend les risques*, ce qui fera apparaître le message suivant :

page 64

> Si vous comprenez ce qui se passe, vous pouvez indiquer à Iceweasel de commencer à faire confiance à l'identification de ce site. Même si vous avez confiance en ce site, cette erreur pourrait signifier que quelqu'un est en train de pirater votre connexion.
>
> N'ajoutez pas d'exception à moins que vous ne connaissiez une bonne raison pour laquelle ce site n'utilise pas d'identification certifiée.
>
> Ajouter une exception...

Il faut maintenant cliquer sur *Ajouter une exception…*

Une fenêtre *Ajout d'une exception de sécurité* s'ouvre alors. Dans celle-ci, on peut trouver des informations intéressantes sur la raison pour laquelle le navigateur n'a pas voulu accepter le certificat, sous le titre *État du certificat*. En cas de certificat auto-signé, on pourra lire par exemple la phrase *Ce site essaie de s'identifier lui-même avec des informations invalides*. Il se peut aussi que la date de validité du certificat soit dépassée, ce qui n'en empêche pas forcément l'usage. Il est en tout cas toujours utile de lire cette partie et de se demander si l'on souhaite continuer au regard de ces informations. En cliquant sur *Voir…* puis sur l'onglet *Détails*, on peut regarder plus en profondeur le certificat, et savoir par exemple qui l'a émis, pour combien de temps, *etc.*

Il faut ensuite revenir à l'onglet *Général* afin d'afficher diverses informations sur le certificat qui est présenté à notre navigateur web, dont son empreinte *SHA1*.

À partir de là, vous pouvez commencer à mettre en place les techniques décrites précédemment pour vérifier l'authenticité de ce certificat.

Une fois la vérification faite, deux possibilités s'offrent vous : vous pouvez choisir de n'utiliser ce certificat que temporairement pour la duréé de votre session en cours jusqu'au redémarrage du navigateur, ou l'installer de façon permanente dans votre navigateur afin de ne plus avoir à faire cette vérification.

Dans le cas du *Tor Browser Bundle*, l'importation permanente d'un certificat devra être reproduit après chaque mise à jour, les fichiers et la configuration du navigateur étant remplacés à chaque fois par la nouvelle version. Il peut donc être utile une fois le certificat vérifié de le sauvegarder dans un endroit sûr, ou d'en imprimer l'empreinte afin de pouvoir effectuer la vérification plus rapidemment les fois suivantes.

Tails ne permet pas pour l'instant de conserver simplement de manière persistante les certificats importés. Par conséquent, on doit à nouveau contrôler ces certificats après chaque extinction. Il est alors conseillé de noter les empreintes *SHA1* quelque part pour les avoir toujours sous la main.

Dans le cas où une utilisation temporaire suffit, taper sur la touche *Echap* pour fermer la fenêtre, puis cliquer sur *Confirmer l'exception de sécurité*.

Si vous souhaitez installer le certificat de façon permanente, cliquer sur *Voir…* puis sur l'onglet *Détails*, et enfin sur le bouton *Exporter*. Enregistrer le certificat sur votre disque dur *en ajoutant* .pem *au nom du fichier*, ce qui facilitera pour la suite son

importation. Si vous utilisez *Tails*, enregistrer ce fichier dans un dossier de la partition persistante, vous pourrez ainsi le ré-importer en toute confiance lors de vos prochaines sessions. Fermer ensuite cette fenêtre puis cliquer sur *Annuler*. Il n'est pas encore temps d'accepter le certificat.

Dans la fenêtre du navigateur, cliquer sur *Édition* puis *Préférences*. Choisissez l'onglet *Avancé*, puis le sous-menu *Certificats*. Enfin cliquez sur *Afficher les certificats*. Choisir l'onglet *Serveurs*, et cliquer ensuite sur *Importer*, puis sélectionner le fichier enregistré à l'étape précédente et confirmer. Sélectionner ensuite le certificat du site qui est apparu dans la liste des certificats connus de votre navigateur, puis cliquer sur le bouton *Modifier la confiance…*. Cliquer sur *Avoir confiance en l'authenticité de ce certificat.*, puis confirmer en cliquant sur *OK*. Le certificat est dorénavant installé de façon permanente dans le navigateur, vous pouvez fermer les fenêtres que nous avons ouvertes, et vous rendre sur le site que vous vouliez visiter.

15.3 Ajouter une autorité de certification

Pour ajouter une AC à la liste de celles reconnues par le navigateur web, il faudra récupérer son certificat racine [2]. Dans Firefox, il faut cliquer sur le lien vers le certificat racine de l'autorité de certification en question. Ce lien est souvent disponible sur le site web de l'AC. Par exemple pour celui de l'autorité de certification CACert il faut cliquer là [http://www.cacert.org/certs/root.crt]. Si celui-ci n'est pas déjà installé, s'ouvre alors une boîte de dialogue :

> On vous a demandé de confirmer une nouvelle autorité de certification (AC).
>
> Voulez-vous faire confiance à « cacert.org » pour les actions suivantes ?
>
> ☑ confirmer cette AC pour identifier les sites Web
>
> ☐ confirmer cette AC pour identifier les utilisateurs de courrier.
>
> ☐ confirmer cette AC pour identifier les développeurs de logiciels.
>
> Avant de confirmer cette AC pour quelque raison que ce soit, vous devriez l'examiner elle, ses méthodes et ses procédures (si possible).

Cliquer maintenant sur « Voir » pour « Examiner le certificat d'AC ».

tome 1 § 5.2

Une fenêtre apparaît alors, nous indiquant qui a émis le certificat, pour qui, sa validité, et en bas sont affichées les empreintes numériques du certificat.

Il est temps à ce moment de mettre en place les techniques décrites précédemment pour vérifier l'authenticité de ce certificat.

Si les empreintes correspondent, on peut alors cliquer sur « Fermer ». Dans la fenêtre précédente, cocher la case « confirmer cette AC pour identifier les sites Web », puis cliquer sur le bouton « OK ».

15.4 Trouver l'empreinte d'un certificat déjà installé

Cette empreinte peut être visualisée en cliquant sur *Édition* dans le menu du navigateur, puis *Préférences*. Choisir l'onglet *Avancé*, puis le sous-menu *Certificats*. Enfin cliquer sur *Afficher les certificats*. Vous pourrez ensuite trouver les certificats des sites installés en choisissant l'onglet *Serveurs*, puis en sélectionnant le site en question dans la liste et en cliquant sur le bouton *Voir*. La même opération est possible pour les autorités de certification en choisissant plutôt l'onget *Autorités*.

2. Wikipédia, 2014, *Root certificate* [https://fr.wikipedia.org/wiki/Root_certificate]

Chapitre 16

Utiliser un clavier virtuel dans Tails

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : quelques minutes.

Nous avons vu dans le premier tome qu'un ordinateur peut être matériellement corrumpu. Il peut notamment contenir des keyloggers matériels qui pourraient enregistrer tout ce qui est tapé sur le clavier. Les textes que l'on écrit, des actions que l'on exécute, mais surtout les mots de passe que l'on saisit.

tome 1 § 3.3

Lorsqu'on a un doute quant à la confiance à accorder à un ordinateur sur lequel on va utiliser *Tails*, il est possible d'utiliser un clavier virtuel afin de rendre inefficace la récupération des frappes sur le clavier. Attention cependant, cette méthode ne protège pas d'un mouchard qui enregistrerait l' affichage de l'écran.

tome 1 ch. 3

16.1 Utiliser un clavier virtuel dans Tails

Une clavier virtuel est un logiciel ayant l'apparence d'un clavier et qui nous permet d'entrer des caractères sans utiliser le clavier matériel de l'ordinateur. Il peut être utilisé avec plusieurs dispositifs de pointage comme une souris, un écran tactile, ou un pavé tactile par exemple.

La clavier virtuel *Florence* est installé par défaut dans *Tails*. Il est automatiquement démarré lors du lancement de *Tails* et est accessible en cliquant sur son icône en forme de clavier dans la zone de notification en haut à droite de l'écran ou à partir de *Applications* → *Accès universel* → *Clavier virtuel Florence*.

Une fois lancé, taper ses mots de passe en utilisant son dispositif de pointage préféré.

Chapitre 17

Utiliser le client mail Claws Mail

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 15 à 30 minutes.

Cette partie va décrire la méthode à adopter pour configurer le client de messagerie Claws Mail de manière à l'utiliser pour toutes tâches ayant trait à ses emails.

17.1 Installer le client mail Claws Mail

Si l'on utilise une Debian chiffrée, il va tout d'abord falloir installer le logiciel Claws Mail. Pour cela, installer les paquets `claws-mail` et `claws-mail-pgpmime` pour le chiffrement.

tome 1 ch. 15
tome 1 § 16.3

Claws Mail ainsi que les paquets nécessaires au chiffrement des emails sont inclus par défaut dans *Tails*. Si l'on ne souhaite pas avoir à reconfigurer Claws Mail à chaque démarrage de *Tails*, ainsi que conserver ses emails, contacts *etc.* d'une session à l'autre, on activera au préalable l'option *Claws Mail* dans la persistance de *Tails*.

page 97
tome 1 § 14.5

17.2 Lancer Claws Mail

Une fois les paquets installés, lancer *Claws Mail* à partir du menu *Applications* → *Internet* → *Claws Mail*.

17.3 Configurer un compte email

Lorsqu'on lance *Claws Mail* et qu'aucun compte email n'y est configuré, une fenêtre propose de nous assister dans la configuration du logiciel, et pour l'ajout d'un premier compte. Si en revanche on veut *ajouter* un compte mail supplémentaire à Claws Mail, aller directement au chapitre « Avec l'outil de création de compte ».

17.3.1 Avec l'assistant (1er démarrage)

Si l'on désire configurer un compte email, cliquer sur *Suivant*. On arrive sur une fenêtre *Informations personnelles*. Remplir a minima le champ *Votre nom* avec le pseudonyme qu'on voudra voir apparaître dans les en-têtes et le champ *Votre adresse email*. Cliquer ensuite sur *Suivant*.

Nous arrivons désormais sur la page de configuration relative à la *Réception du courrier*. De la même manière, remplir les champs nécessaires avec les informations correspondantes, à commencer par le protocole email que l'on préfère utiliser dans *Type*

page 97

page 97

de serveur, à savoir IMAP ou POP. Les informations correspondant à l'*Adresse du serveur* et aux méthodes de chiffrement de la connexion acceptées se trouvent en général sur le site web du fournisseur d'email. Vous pouvez également les trouver en utilisant un moteur de recherche, avec des mots-clés tels que, par exemple pour le fourniseur d'email Riseup, "riseup configure client mail".

Remplir ensuite les champs *Utilisateur*, *Mot de passe* et éventuellement choisir quelques options concernant le chiffrement de la connexion. Une fois la partie *Réception du courrier* terminée, cliquer sur *Suivant*.

Configurons maintenant l'*Envoi du courrier*.

Tout comme pour la réception du courrier, spécifier l'*Adresse du serveur d'envoi (SMTP)* ainsi que le *Nom d'utilisateur SMTP* et le *Mot de passe SMTP*. Ceux-ci sont généralement identiques à l'*Utilisateur* et au *Mot de passe* indiqués auparavant pour la *Réception du courrier*.

Cliquer ensuite sur *Suivant*. On nous demande alors de choisir un dossier dans lequel les emails seront stockés : on peut ici laisser le choix par défaut. Cliquer une dernière fois sur *Suivant*.

Une notification nous indique que la configuration est terminée, il ne nous reste plus qu'à cliquer sur *Enregistrer*.

Claws Mail est désormais prêt à réceptionner les messages. Si vous souhaitez dès maintenant ajouter un compte supplémentaire, continuez la lecture. Sinon, vous pouvez passer directement au chapitre suivant, consacré à la configuration avancée de Claws Mail.

17.3.2 Avec l'outil de création de compte

Pour ajouter un nouveau compte à *Claws Mail*, sélectionner *Création d'un nouveau compte..* depuis le menu *Configuration* de *Claws Mail*.

Une fenêtre *Configuration d'un nouveau compte* s'ouvre alors. Sur la gauche est affichée une liste de préférences. Dans la partie *Général* de *Compte*, remplir les informations nécessaires au fonctionnement du compte. Dans les *Informations personnelles*, fournir le *Nom complet* (qui peut être la partie située avant le @ de l'adresse email), ainsi que l'*Adresse email*. Dans la partie *Configuration de serveurs*, choisir le *Protocole* pour la réception du courrier, écrire les adresses du *Serveur de réception* et du *Serveur d'envoi (SMTP)*. Les adresses de ces serveurs, ainsi que divers détails de configuration, se trouvent facilement sur les sites web des hébergeurs d'email en question. Enfin fournir le *Nom d'utilisateur* et le *Mot de passe* associé. Passer ensuite à la partie *Envoyer* des *Préférences* de *Compte* et cocher *Authentification SMTP (SMTP AUTH)* dans la partie *Authentification*. Cliquer enfin sur *Valider*.

De nombreuses autres options de configurations sont disponibles, que nous détaillerons en partie dans le paragraphe suivant.

17.4 Configuration avancée de Claws Mail

Une fois *Claws Mail* configuré pour un compte email, on peut vouloir optimiser sa configuration, pour qu'elle nous soit plus agréable ou pour qu'elle réduise les risques en termes de sécurité informatique.

Pour cela, choisir *Édition des comptes…* dans le menu *Configuration* de *Claws Mail*. Une fenêtre *Édition des comptes* s'ouvre : sélectionner d'un clic le compte email à éditer, puis cliquer sur *Modifier*. Nous n'allons pas faire une tour exhaustif des options de configuration, mais de quelques-unes qui nous semblent utiles.

Tout d'abord, si l'on a choisi d'utiliser le protocole POP, dans la partie *Réception* des *Préférences* de *Compte*, on peut décider de la durée après laquelle les messages seront

supprimés des serveurs après rapatriement. Cela est bien sûr sans grande garantie et dépend notamment notre hébergeur mail : nous ne pouvons qu'espérer qu'il efface véritablement nos données.

tome 1 § 4.3

Ensuite, dans la partie *SSL* des *Préférences* de *Compte*, si ce n'est pas déjà sélectionné, choisir d'*Utiliser SSL pour les connexions POP3* ou *Utiliser SSL pour les connexions IMAP4* en fonction du protocole choisi ainsi que *Utiliser SSL pour les connexions SMTP*. Si le *certificat* du serveur n'est pas signé par une Autorité de Certification, il faudra le vérifier et l'ajouter manuellement. Enfin, dans la partie *Avancé* des *Préférences* de *Compte*, il est possible que les ports des protocoles utilisés ne soient pas les bons avec les réglages par défaut. Si c'est le cas, modifier *Port SMTP*, *Port IMAP4* ou *Port POP3* en fonction des informations de configurations disponibles chez notre hébergeur email.

page 121

17.4.1 Activer le plugin de chiffrement PGP/MIME

Si l'on désire utiliser la cryptographie asymétrique, que ce soit pour chiffrer des emails, les signer ou les deux, il faut configurer le logiciel *Claws Mail* selon nos préférences.

Pour cela, ouvrir la fenêtre d'*Édition des comptes* en choisissant *Édition des comptes…* depuis le menu *Configuration* de *Claws Mail*. Sélectionner d'un clic le compte email à éditer, puis cliquer sur le bouton *Modifier*. Aller dans la partie *Confidentialité* des *Préférences* de *Compte*. Dans le menu déroulant attenant à *Système de confidentialité par défaut*, choisir *PGP MIME*. Ce mode de chiffrement permet de chiffrer le contenu ainsi que les pièces jointes éventuelles de l'email. Il est possible ensuite de décider de règles par défaut quant au chiffrement automatique de réponses à des emails chiffrés, parmi d'autres options. Faites le tour, et en cas de doute, faites une recherche sur Internet ou demandez autour de vous. Une fois votre choix effectué, cliquez sur le bouton *Valider*.

Il est également possible de modifier quelques options concernant la gestion du chiffrement dans *Claws Mail* en allant dans *Configuration* → *Préférences...* une fenêtre *Préférences* s'ouvre dans laquelle on peut aller voir du côté de la rubrique *GPG* de la partie *Modules* des *Préférences*. Il sera par exemple possible d'activer ou non la mémorisation de la phrase de passe d'une paire de clés, et si oui, pendant un temps donné.

Chapitre 18

Utiliser OpenPGP

Le standard Internet [1] OpenPGP est un format de cryptographie qui permet notamment d'effectuer et de vérifier des signatures numériques ainsi que de chiffrer et de déchiffrer des emails ou des fichiers.

page 141

Nous allons ici détailler différents outils de cryptographie ayant en commun l'usage d'OpenPGP.

page 139
page 138
page 139

18.1 Importer une clé OpenPGP

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : quelques minutes.

Le but de ce chapitre est d'importer une clé OpenPGP, que nous utiliserons pour vérifier des signatures numériques ou pour chiffrer des messages. La procédure est la même sous *Tails* ou avec une Debian chiffrée.

Ouvrir *Mots de passe et clés de chiffrement* depuis le menu *Applications* → *Outils système* → *Préférences*.

Importer une clé ne signifie pas avoir vérifié qu'elle appartient bien au propriétaire supposé. Nous verrons par la suite qu'il faut pour cela effectuer d'autres opérations, comme étudier ses signatures ou son empreinte numérique.

18.1.1 Afficher les clés disponibles

Pour afficher les clés importées, cliquer sur *Affichage* → *Tout afficher*. Choisir *Affichage* → *Par trousseau* permet de mieux s'y retrouver.

18.1.2 Si l'on dispose de la clé dans un fichier

Cliquer sur *Fichier* → *Importer…*. dans la fenêtre qui s'ouvre, sélectionner le fichier contenant la clé, puis cliquer sur *Ouvrir*. Une fenêtre affiche des informations sur la clé. Si c'est bien la clé qu'on souhaite importer, cliquer sur *Importer*.

18.1.3 Si l'on veut chercher la clé en ligne

Toujours dans la fenêtre *Mots de passe et clés de chiffrement*, cliquer sur *Distant* → *Chercher des clés distantes…*

1. Wikipédia, 2014, *Standard Internet* [https://fr.wikipedia.org/wiki/Standard_Internet]

Dans la fenêtre qui s'ouvre alors, taper un nom, un numéro de clé ou toute autre information permettant de trouver la clé recherchée, par exemple : « 0x63FEE659 », « 63FEE659 » ou « Alice Dupont ». Cliquer ensuite sur *Chercher*.

Une fenêtre de résultats s'ouvre. Il peut d'ailleurs y avoir de nombreux noms correspondant à notre recherche, comme par exemple pour *Erinn Clark*. Lequel choisir ? Si l'on sait que la clé que l'on cherche a comme identifiant 63FEE659, on fera clic-droit sur l'un des résultats puis *Propriétés*. On pourra alors comparer l'empreinte de la clé sélectionnée avec celle désirée. Une fois la bonne clé trouvée, la sélectionner *Fichier* → *Importer*, puis fermer la fenêtre.

La clé importée devrait être visible dans le trousseau de *Clés PGP*.

18.2 Vérifier l'authenticité d'une clé publique

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : de quelques minutes à une demi-heure.

Lors de l'utilisation de la cryptographie asymétrique, il est crucial de s'assurer que l'on dispose la véritable clé publique de notre correspondant. Sinon, on s'expose à une attaque de l'homme du milieu : on authentifie ou on chiffre bien notre correspondance… pour notre adversaire.

page 63
page 64

On devra tout d'abord choisir une méthode pour s'assurer que l'on dispose de la bonne clé publique. On indiquera ensuite à OpenPGP notre confiance en cette clé.

18.2.1 Établir une confiance

tome 1 ch. 6

En fonction des exigences de notre modèle de risque et de nos possibilités, on pourra choisir différentes façons pour vérifier l'authenticité d'une clé publique. Admettons qu'on doive vérifier l'authenticité de la clé publique d'Alice.

Se transmettre la clé par un canal sûr

Lorsque c'est possible, le plus simple est de se passer en main propre, à l'aide d'une clé USB par exemple, le fichier contenant la clé publique. Alice exporte alors sa clé publique vers un fichier, qu'elle stocke sur une clé USB éventuellement chiffrée qu'elle nous donne ensuite. On importera ensuite directement la clé publique d'Alice à partir de ce fichier.

page 137
tome 1 ch. 18
page préc.

Se transmettre l'empreinte par un canal sûr

L'un des inconvénients de la méthode précédente est qu'elle nécessite de se passer un fichier informatique par un moyen sûr. Cela n'est pas toujours possible. Heureusement, ce n'est en fait pas nécessaire : il suffit d'obtenir, par un moyen sûr, une somme de contrôle de la clé publique, qu'on appelle « empreinte » (ou « fingerprint » en anglais).

tome 1 § 5.2

Alice peut ainsi publier sa clé publique sur Internet, par exemple sur son blog ou sur un serveur de clés. De notre côté, nous téléchargeons cette clé de façon non authentifiée, puis on vérifie que l'empreinte de la clé correspond à celle qu'Alice nous a fait parvenir de façon *authentifiée*.

Que gagnons-nous à utiliser cette méthode ? Au lieu de devoir se faire passer un fichier, il est suffisant de se transmettre une ligne de caractères comme celle-ci :

```
0D24 B36A A9A2 A651 7878 7645 1202 821C BE2C D9C1
```

Par exemple, Alice, qui est une personne organisée, peut avoir en permanence sur elle un exemplaire de l'empreinte de sa clé publique écrite sur un bout de papier. Il nous suffit alors de la croiser pour qu'elle nous la passe : pas besoin d'ordinateur ni de clé USB.

Si l'on ne peut pas rencontrer Alice, elle pourra aussi nous envoyer cette empreinte par courrier postal, et on pourra l'appeler pour qu'elle nous la lise par téléphone. La vérification sera moins bonne qu'en se voyant directement, mais il reste plus difficile pour un adversaire de nous envoyer un courrier postal avec sa clé et de répondre au numéro de téléphone d'Alice en nous lisant son empreinte, tout en imitant sa voix.

Ça se complique encore si on ne connaît pas Alice. Dans ce cas, il nous faudra faire confiance à des personnes qui prétendent la connaître. Encore une fois, il n'y a pas de recette magique, mais combiner différents moyens de vérification permet de compliquer la tâche d'un éventuel adversaire souhaitant monter une "attaque de l'homme du milieu" : il nous est possible de demander à plusieurs personnes qui prétendent connaître Alice plutôt qu'à une seule ; utiliser plusieurs moyens de communication différents, *etc.*

page 64

Utiliser les toiles de confiance

OpenPGP intègre la notion de *confiance transitive* avec les toiles de confiance. Une fois la clé d'Alice téléchargée, on peut lister les identités qui ont signé sa clé : ces personnes déclarent publiquement avoir vérifié que cette clé appartient bien à Alice. Si l'on connaît une de ces personnes, ou un tiers qui a confiance en une de ces personnes, OpenPGP peut établir des chemins de confiance entre les identités auxquelles on fait confiance et d'autres avec lesquelles on souhaite communiquer.

page 66

18.3 Signer une clé

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : quelques minutes.

Une fois qu'on a établi une confiance en la clé d'Alice, il est utile d'informer OpenPGP qu'il peut faire confiance à sa clé. Cette opération s'appelle *Signer* une clé. La procédure est la même sous *Tails* ou avec une Debian chiffrée.

Deux options s'offrent à nous :

- signer la clé d'Alice localement, ce qui permet de ne pas publier que notre identité est « liée » à celle d'Alice ;
- signer la clé d'Alice publiquement, ce qui permet à n'importe quel utilisateur de la toile de confiance de profiter des vérifications qu'on a faites.

Encore une fois, pas de bonne réponse, mais un choix à faire en fonction de nos besoins et de notre modèle de risques.

Pour signer une clé, rendez vous dans *Mots de passes et clés*, qui est accessible à partir du menu *Applications* → *Outils système* → *Préférences*.

Pour voir les clés OpenPGP cliquer sur le menu *Affichage* → *Tout afficher* et *Affichage* → *Par trousseau*.

Si la clé n'est pas présente, l'importer.

page 131

Une fois la clé d'Alice repérée dans la fenêtre principale, double-cliquer dessus pour afficher les détails de la clé. Vérifier que c'est la bonne clé, par exemple en vérifiant

son empreinte dans l'onglet *Détails*. Choisir ensuite l'onglet *Confiance* puis cliquer sur *Signer la clé*.

Choisir avec quelle précaution on a vérifié la clé, par exemple *superficiellement* si on a vérifié l'empreinte par téléphone, ou *très sérieusement* si on connaît bien Alice et qu'elle nous a donné sa clé ou son empreinte en main propre.

En bas de la fenêtre, cliquer sur *Les autres ne peuvent pas voir cette signature* si l'on souhaite cacher les liens entre notre identité et Alice.

Cliquer ensuite sur *Signer*, et saisir la phrase de passe de notre clé privée dans la boîte de dialogue qui s'affiche.

OpenPGP sait maintenant qu'on a confiance en la clé d'Alice.

18.4 Créer et maintenir une paire de clés

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 15 minutes à une heure.

Nous allons détailler dans cet outil la création et une partie de la gestion d'une paire de clés de chiffrement. Il est bon de rappeler quelques notions de base à toujours avoir à l'esprit. Tout d'abord le fait que toutes les clés de chiffrement n'utilisent pas le même algorithme. Nous avons parlé du chiffrement RSA mais il en existe plusieurs autres. Et si des clés de chiffrement utilisent en effet le même algorithme, elles ne sont pas pour autant de même taille. De plus, certaines ont des dates d'expirations, à laquelle elles périment, d'autres n'en ont pas.

page 61

18.4.1 Créer une paire de clés

Afin de créer une paire de clés, lancer *Mots de passe et clés* à partir du menu *Applications → Outils système → Préférences*.

Dans la fenêtre qui s'ouvre alors, cliquer sur le bouton *Nouveau…* dans le menu *Fichier*. Sélectionner ensuite *Clé PGP* puis cliquer sur *Continuer*.

page 53

Une nouvelle fenêtre s'ouvre. Entrer un *Nom complet* correspondant à l'identité contextuelle utilisée, ainsi que l'*Adresse électronique* qui lui est associée. Il est possible de mettre l'identifiant de l'adresse email se situant avant le symbole @ comme *Nom complet*. Cliquer ensuite sur *Options avancées de clé* pour choisir la taille de la clé et sa date d'expiration. Le *Type de chiffrement* par défaut est *RSA*. Le laisser tel quel. La *Force de la clé* proposée par défaut, 2048 bits, est considéré comme sûre jusqu'en 2020 [2]. On peut choisir la force de la clé la plus élevée disponible, à savoir 4096 bits, si l'on souhaite protéger ses communications plus fortement ou plus longtemps. Il est conseillé de choisir une *Date d'expiration* pour la clé. Si c'est la première fois que l'on crée une paire de clés, on choisira une date d'expiration comprise entre 1 an et 2 ans par exemple. Cliquer enfin sur *Créer*.

tome 1 ch. 12

Une nouvelle fenêtre s'ouvre, demandant une phrase de passe pour protéger la clé. C'est le moment de choisir une bonne phrase de passe puis de la taper deux fois, avant de cliquer sur *Valider*. Attention cependant à ne pas confondre la phrase de passe que l'on donne ici avec une des clés de la paire de clés de chiffrement. La phrase de passe sert uniquement à pouvoir restreindre l'utilisation la clé privée de notre paire.

2. Agence nationale de la sécurité des systèmes d'information, 2010, *Mécanismes cryptographiques – Règles et recommandations concernant le choix et le dimensionnement des mécanismes cryptographiques* [http://www.ssi.gouv.fr/IMG/pdf/RGS_B_1.pdf]

Une fenêtre *Génération de clé* affiche alors une barre de progression. Cela peut prendre plusieurs minutes. C'est le moment de faire bouger sa souris, d'utiliser son clavier ou encore d'utiliser le disque dur si cela est possible, afin d'aider son ordinateur à génerer des données aléatoires. Celles-ci sont nécessaires au processus de génération de la clé [3].

Cette étape de création de clés effectuée, il est bon de penser à la manière de sauvegarder notre paire de clés. Étant en partie secrètes, il s'agit de ne pas les laisser traîner n'importe où. La clé privée doit être uniquement accessible à la personne supposée y avoir accès. Le mieux est de conserver cette paire de clés sur un volume chiffré, que celui-ci soit une clé USB, un disque dur interne ou externe, ou la persistance de *Tails*.

tome 1 ch. 18
tome 1 § 14.5

18.4.2 Exporter sa clé publique

Pour qu'une personne puisse nous envoyer des emails chiffrés, elle doit disposer de notre clé publique. Pour cela il va falloir l'exporter du logiciel *Mots de passe et clés* afin de la transmettre à nos correspondants.

Voir comment exporter une clé.

page 137

18.4.3 Publier sa clé publique sur les serveurs de clés

Si l'existence de l'identité contextuelle à laquelle correspond la clé n'est pas elle-même confidentielle, on pourra publier notre clé publique sur un serveur de clés, afin que quiconque désirant nous envoyer des emails chiffrés puisse la télécharger à cette fin. Pour cela, cliquer sur *Sychroniser et publier des clés…* dans le menu *Distant*. Une fenêtre *Synchroniser les clés* apparaît.

page 53

Si elle affiche *Aucun serveur de clés n'a été choisi pour publier, vos clés ne seront donc pas mises à disposition des autres*, cliquer sur *Serveurs de clés* et choisir un serveur dans le menu déroulant en face de *Publier les clés sur :*, puis cliquer sur *Fermer*.

Cliquer alors sur *Synchroniser* pour publier la clé.

18.4.4 Obtenir l'empreinte d'une clé

Si l'on transmet notre clé publique par un moyen non authentifié (par exemple un courrier électronique non signé), il peut être utile de faire parvenir à notre correspondant l' empreinte de notre clé par un moyen authentifié, afin qu'il s'assure de son intégrité. L'empreinte est accessible dans l'onglet *Détails* disponible en double-cliquant sur une clé. On pourra par exemple la noter sur un papier qu'on donnera en main propre à notre correspondant.

tome 1 § 5.2.2

18.4.5 Générer un certificat de révocation et le conserver à l'abri

Si un adversaire mettait la main sur notre clé privée, ou simplement si on la perdait, il est nécessaire de la *révoquer*, afin que nos correspondants soient au courant qu'il ne faut plus l'utiliser. On crée pour cela un *certificat de révocation*.

Il est conseillé de créer le certificat de révocation immédiatement après la paire de clés, car si l'on perd la clé ou qu'on oublie sa phrase de passe, il ne nous sera plus possible de créer de certificat de révocation.

Le certification se présente sous la forme d'un fichier ou de quelques lignes de « texte », qu'il nous faudra stocker dans un endroit sûr, par exemple sur une clé USB chiffrée, chez une personne de confiance ou sur un papier bien caché. En effet, toute personne qui a accès à ce fichier peut révoquer notre paire de clés, et donc nous empêcher de communiquer.

Pour générer le certificat, on doit malheureusement utiliser un terminal.

tome 1 ch. 11

3. Zvi Gutterman, Benny Pinkas, Tzachy Reinman, 2006, *Analysis of the Linux Random Number Generator* [http://www.pinkas.net/PAPERS/gpr06.pdf] (en anglais).

On va commencer la commande en tapant (**sans** faire *Entrée*) :

```
gpg --gen-revoke
```

Puis taper l'identifiant de notre clé, accessible dans l'onglet *Propriétaire*, disponible en double-cliquant sur la clé.

Cela devrait donner quelque chose comme :

```
gpg --gen-revoke 2A544427
```

Appuyer alors sur la touche *Entrée* pour lancer la commande.

GnuPG nous pose alors quelques questions :

```
sec  2048R/2A544427 2013-09-24 Alice (exemple seulement) <alice@example.org>

Générer un certificat de révocation pour cette clé ? (o/N)
```

Taper *o* puis cliquer sur *Entrée*. Le terminal nous renvoie ensuite :

```
sec  2048R/2A544427 2013-09-24 Alice (exemple seulement) <alice@example.org>

Générer un certificat de révocation pour cette clé ? (o/N) o
choisissez la cause de la révocation :
  0 = Aucune raison spécifiée
  1 = La clé a été compromise
  2 = La clé a été remplacée
  3 = La clé n'est plus utilisée
  Q = Annuler
(Vous devriez sûrement sélectionner 1 ici)
Votre décision ?
```

Nous préparons un certificat pour le cas où notre clé soit compromise. Nous allons donc taper le chiffre *1*, puis appuyer sur la touche *Entrée*.

GnuPG nous demande alors une description de problème :

```
Entrez une description optionnelle ; terminez-la par une ligne vide :
>
```

On ne sait pas, puisque la clé n'est pas encore compromise, et on va donc simplement accepter une description vide en appuyant à nouveau sur la touche *Entrée*.

GnuPG nous demande alors confirmation :

```
Cause de révocation :La clé a été compromise
(Aucune description donnée)
Est-ce d'accord ? (o/N)
```

Taper *o* puis appuyer sur la touche *Entrée* pour accepter. *GnuPG* nous demande alors la phrase de passe associée à cette paire de clés, puis affiche le certificat de révocation :

```
sortie avec armure ASCII forcée.
Certificat de révocation créé.

Veuillez le déplacer sur un support que vous pouvez cacher ; toute personne
accédant à ce certificat peut l'utiliser pour rendre votre clef inutilisable.
Imprimer ce certificat et le stocker ailleurs est une bonne idée, au cas où le
support devienne illisible. Attention quand même  :le système d'impression
utilisé pourrait stocker ces données et les rendre accessibles à d'autres.
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version :GnuPG v1.4.10 (\mbox{GNU/Linux})
Comment :A revocation certificate should follow

iQEfBCABCgAJBQJSQZVMAh0CAAoJEMYS/iAqVEQnzFsH/3NMzeXy0XbOJ3Q+g2mA
xEAl4G8VesEYDE8LHzemNmkyrrMKNGpllPJVkyMXKBLYTojQjjL6QhL1nyqaUavs
eOmaa1Swa9PgI6AJZrkmiMk74CCXJqQDb5uupZNQ3UsoGHqKcirYUHyOeEQ/m94Q
xMaPjpCMi9tIJjnb1T8svDuwhpsh2GjZhOuyUedyyD4r/noT8YYhWKNC98ElPQkH
VVEzu6TJu0IKRp7OJgPCb8cJ6odsm3jPxjIF+f/cz9WIud8EB3HJVIxoMm183XI+
HtddcOxSsdIljuk6ddqgyQDTPJVex+EYdG0FreT7OrFzKXo316/4RSWKX/klshSp
O/8=
=cpvr
-----END PGP PUBLIC KEY BLOCK-----
```

Le certificat est la partie située entre `BEGIN PGP PUBLIC KEY BLOCK` et `END PGP PUBLIC KEY BLOCK`. C'est à conserver à l'abri. On va commencer par la sélectionner et la copier dans le presse-papiers (clic-droit, puis *Copier*), puis par ouvrir l'*Éditeur de texte gedit* (accessible depuis le menu *Applications* puis *Accessoires*), et la coller dans un nouveau document (clic droit, puis *Coller*).

Selon notre choix, on pourra :

- l'**enregistrer** avec la commande *Enregistrer* du menu *Fichier*. Choisir une nom de fichier clair et le terminer par `.rev`. Par exemple `Certificat de révocation pour la clé 2A544427.rev` ;
- l'imprimer avec la commande *Imprimer…* du menu *Fichier*.

Si notre clé venait à être compromise, on utiliserait ce certificat pour la révoquer. page 142

18.4.6 Effectuer la transition vers une nouvelle paire de clés

Avant que notre paire de clés expire, ou lorsque des avancées dans le domaine de la cryptographie nous obligent à utiliser des clés plus sûres, il nous faudra créer une nouvelle paire de clés.

On suivra pour cela les instructions ci-dessus.

On prendra ensuite soin de signer notre nouvelle clé avec l'ancienne en suivant la section « Signer une clé » de l'outil vérifier l'authenticité d'une clé. On exportera alors notre nouvelle clé et on la fera parvenir aux personnes avec lesquelles on communique. page 133

Quelques mois plus tard, on pourra révoquer notre ancienne clé. page 142

18.5 Exporter une clé publique OpenPGP

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : quelques minutes.

Le but de cet outil est d'exporter une clé OpenPGP, utilisée par exemple pour la transmettre à nos contacts afin qu'ils puissent nous écrire, ou pour vérifier des signatures numériques. La procédure est la même sous *Tails* ou avec une Debian chiffrée.

Ouvrir *Mots de passe et clés* depuis le menu *Applications* → *Outils système* → *Préférences*.

18.5.1 Afficher les clés disponibles

Pour afficher les clés importées, cliquer sur *Affichage* → *Tout afficher*. Choisir *Affichage* → *Par trousseau* permet de mieux s'y retrouver.

18.5.2 Pour exporter la clé vers un fichier

Le fichier que l'on va exporter contiendra notre clé publique, nécessaire aux personnes qui souhaitent nous envoyer des emails chiffrées. Aller dans *Mots de passe et clés*, sélectionner notre clé OpenPGP et choisir *Exporter…* dans le menu *Fichier*. En bas à droite de la fenêtre choisir *Clés blindées PGP* dans le menu déroulant à la place de *Clés PGP*. Choisir un emplacement d'exportation et un nom de fichier, puis cliquer sur *Exporter*.

18.5.3 Pour exporter la clé vers un serveur de clés

page 134 Pour cela voir comment publier sa clé publique sur les serveurs de clés.

18.6 Utiliser la crytographie asymétrique pour chiffrer ses emails

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : quelques minutes.

Nous allons explorer l'usage de la crytographie asymétrique dans le cas particulier du chiffrement d'emails.

Selon qu'on utilise un client mail ou le webmail, la méthode à employer pour chiffrer ses emails sera différente.

18.6.1 Chiffrer ses emails dans Claws Mail

page 127 Une fois Claws Mail démarré et configuré, cliquer sur le bouton *Composer* afin de débuter la rédaction d'un nouveau message. Une fenêtre *Composition d'un message* s'ouvre dans laquelle on va rédiger son email. Avant ou après rédaction de son email, mais en tout cas avant de l'envoyer, choisir *Chiffrer* dans le menu *Options* de la fenêtre *Composition d'un message*. Une fois notre email terminé, cliquer sur *Envoyer*, un *Avertissement pour le chiffrement* nous prévient du fait que les en-têtes ne seront pas chiffrés. Éventuellement, une fenêtre *Faites-vous confiance à ces clés ?* peut s'afficher, y répondre en conséquence. Si l'adresse email de destinataire correspond à une clé publique (et une seule) présente dans le logiciel *Mots de passe et clés*, elle sera choisie automatiquement pour chiffrer l'email. Sinon une fenêtre *Sélection de clés* s'ouvre pour que nous en choisissions une.

18.6.2 Chiffrer ses emails pour un webmail avec *Tails*

page 26 Si l'on préfère chiffrer ses emails en utilisant un webmail, faire comme suit, tout d'abord, éviter de rédiger son message dans la fenêtre du navigateur web pour le chiffrer ensuite. En effet, certaines attaques, notamment *via* JavaScript, sont susceptibles

d'accéder à votre texte depuis ce même navigateur. Il serait fort regrettable d'offrir en clair un texte que l'on souhaite chiffrer. tome 1 § 5.1.1

On ne va *pas* expliquer comment chiffrer ses emails pour un webmail avec une Debian chiffrée, mais uniquement avec *Tails*.

La méthode actuellement conseillée pour chiffrer un email, de même que pour chiffrer un texte, est décrite dans la documentation de *Tails*.

Une fois *Tails* démarré, afficher le bureau et cliquer sur l'icône *Documentation de Tails*. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Chiffrement et vie privée* et cliquer sur la page *Chiffrer et signer du texte avec une clé publique*. Suivre cette page de documentation. tome 1 § 14.4

18.7 Déchiffrer des emails

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : quelques minutes.

Après avoir choisi une méthode de gestion de ses emails, vu comment créer et maintenir une paire de clés ainsi que comment chiffrer des emails, voyons comment les déchiffrer. Là encore, plusieurs méthodes existent en fonction des outils utilisés. page 93 page 134 page ci-contre

18.7.1 Déchiffrer ses emails dans Claws Mail

Pour déchiffrer un email dans le logiciel Claws Mail. Cliquer sur cet email dans *Claws Mail*. Une fenêtre *Saisissez la phrase de passe* s'ouvre, dans laquelle il faudra taper la phrase de passe qui permet d'utiliser la clé privée pour laquelle le message a été chiffré. page 127

Si l'on ne possède pas la clé privée pour laquelle le message a été chiffré, en cliquant dessus, le texte s'affichera dans sa forme chiffrée sans nous demander de phrase de passe.

18.7.2 Déchiffrer ses emails pour un webmail avec *Tails*

Nous allons uniquement expliquer comment déchiffrer ses emails pour un webmail avec *Tails*.

De la même manière que pour chiffrer un email, il faut éviter de le déchiffrer dans le fenêtre du webmail. Des attaques JavaScript sont susceptibles d'accéder au texte depuis le navigateur web utilisé. page 26

Une fois *Tails* démarré, afficher le bureau et cliquer sur l'icône *Documentation de Tails*. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Chiffrement et vie privée* et cliquer sur la page *Déchiffrer et vérifier du texte*. Suivre cette page de documentation. tome 1 § 14.4

18.8 Vérifier une signature numérique OpenPGP

Durée : quelques minutes.

L'objectif de cet outil est de vérifier l'autenthenticité d'un fichier disposant d'une signature numérique OpenPGP. page 62

Les auteurs de ce guide n'ont pour l'instant pas trouvé d'outil graphique permettant d'effectuer des vérifications de signature de façon sérieuse qui soit inclus à la fois dans *Tails* et dans la version actuelle de Debian.

tome 1 ch. 11

Nous allons donc ouvrir un terminal pour faire ces vérifications.

On va commencer la commande en tapant (**sans** faire *Entrée*) :

```
gpg --verify
```

Ajoutez un espace à la suite, puis nous allons cliquer sur l'icône du fichier de signature (souvent suffixé de `.sig` ou `.asc`) et la faire glisser dans le terminal. Après avoir relâché le bouton, ce qui est affiché doit ressembler à :

```
gpg --verify '/home/amnesia/tails-i386-0.20.iso.sig'
```

Attraper alors l'icône du fichier à vérifier et la faire glisser aussi dans le terminal. Après avoir relâché le bouton, ce qui est affiché doit ressembler à :

```
gpg --verify '/home/amnesia/tails-i386-0.20.iso.sig'
↪    '/home/amnesia/tails-i386-0.20.iso'
```

Appuyer alors sur la touche *Entrée* pour lancer la vérification. Elle peut prendre plusieurs minutes en fonction de la taille du fichier et de la puissance de l'ordinateur qu'on utilise. Une fois la vérification terminée, l'ordinateur devrait afficher quelque chose qui ressemble à :

```
gpg :Signature faite le mer. 07 août 2013 19 :36 :23 UTC avec la clef RSA
↪    d'identifiant BE2CD9C1
gpg :Bonne signature de « Tails developers (signing key) <tails@boum.org> »
gpg :                alias « T(A)ILS developers (signing key) <amnesia@boum.org> »

w
```

Ces quelques lignes pourront êtres suivies de quelque chose comme :

```
gpg :Attention  :cette clef n'est pas certifiée avec une signature de confiance.
gpg :            Rien n'indique que la signature appartient à son propriétaire.
Empreinte de clef principale  :0D24 B36A A9A2 A651 7878  7645 1202 821C BE2C D9C1
```

page 132

Ces lignes ne nous indiquent pas que la signature est invalide, mais seulement que l'on a pas encore vérifié l'authenticité de la clé publique de la personne signataire des données, et qu'un adversaire pourrait effectuer une attaque de l'homme du milieu.

page 64

Si la signature était mauvaise, l'ordinateur afficherait quelque chose comme :

```
gpg :Signature faite le mer. 07 août 2013 19 :36 :23 UTC avec la clef RSA
↪    d'identifiant BE2CD9C1
gpg :MAUVAISE signature de « Tails developers (signing key) <tails@boum.org> »
```

18.9 Signer des emails

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : quelques minutes.

Comme expliqué dans la partie concernant les signatures numériques, on peut vouloir assurer l'authenticité d'un message. Nous allons donc désormais voir comment signer numériquement des emails afin de fournir a minima une assurance de leur intégrité, et, au mieux, une assurance quant à leur authenticité.

page 62
tome 1 ch. 5

Depuis Claws Mail

Une fois Claws Mail démarré et configuré, cliquer sur le bouton *Composer* afin de débuter la rédaction d'un nouveau message. Une fenêtre *Composition d'un message* s'ouvre dans laquelle on va rédiger l'email. Avant ou après rédaction de l'email, mais en tout cas avant de l'envoyer choisir *signer* dans le menu *Options* de la fenêtre *Composition d'un message*. Une fois notre email terminé, cliquer sur *Envoyer*. Une fenêtre *Phrase secrète* s'ouvre, nous demandant de saisir la phrase de passe associée à la paire de clés qui va servir à signer le message. Taper la phrase de passe puis cliquer sur *Valider*.

page 127

Depuis un webmail

Il n'existe pas d'outils que l'on puisse recommander pour signer des emails pour un webmail avec une Debian chiffrée. Afin de signer ses emails en utilisant un webmail dans *Tails*, suivre la procédure qui suit.

La méthode actuellement conseillée pour chiffrer un email, de même que pour chiffrer un texte, est décrite dans la documentation de *Tails*.

Une fois *Tails* démarré, afficher le bureau et cliquer sur l'icône *Documentation de Tails*. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Chiffrement et vie privée* et cliquer sur la page *Chiffrer et signer du texte avec une clé publique*. Suivre cette page de documentation.

tome 1 § 14.4

18.10 Signer des données

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : quelques minutes.

L'objectif de cet outil est de signer numériquement des données. Cela peut servir à authentifier l'auteur d'un message ou d'un document, vérifier des logiciels, etc.

page 62

18.10.1 Signer du texte avec *Tails*

Cette méthode ne fonctionne que pour signer du texte avec *Tails*. Pour signer un autre type de ficher ou si l'on utilise une Debian chiffrée, suivre la section suivante.

Une fois *Tails* démarré, afficher le bureau et cliquer sur l'icône *Documentation de Tails*. Dans le menu à droite, cliquer sur *Documentation*. Dans l'index qui s'ouvre, chercher la section *Chiffrement et vie privée*, cliquer sur la page *Chiffrer et signer du texte avec une clé publique* et suivre cette page de documentation.

tome 1 § 14.4

Signer un autre type de fichier

Les auteurs de ce guide n'ont pour l'instant pas trouvé d'outil graphique permettant d'effectuer de signatures numériques de façon sérieuse qui soit inclus à la fois dans *Tails* et dans la version actuelle de Debian.

tome 1 ch. 11

Nous allons donc ouvrir un terminal pour faire ces vérifications. La démarche est la même sous *Tails* et sous Debian.

On va commencer la commande en tapant (**sans** faire *Entrée*) :

```
gpg --detach-sign
```

Nous allons ensuite cliquer sur l'icône du fichier à signer et la faire glisser dans le terminal. Après avoir relâché le bouton, ce qui est affiché doit ressembler à :

```
gpg --detach-sign '/home/amnesia/monfichier.gpg'
```

Dans notre exemple le fichier a comme extension .gpg, mais il peut tout aussi bien s'agir d'un fichier .mp3, .jpg ou autre.

Appuyer alors sur la touche *Entrée* pour effectuer la signature.

L'ordinateur nous demande la phrase de passe de notre clé secrète.

Le processus de signature peut prendre jusqu'à plusieurs minutes en fonction de la taille du fichier et de la puissance de l'ordinateur qu'on utilise. Une fois la signature terminée, elle se présente sous la forme d'un petit fichier ayant le même nom que le fichier original, mais se terminant par l'extension `.sig`, situé au même endroit que le fichier original et qu'il nous faudra transmettre à notre interlocuteur avec celui-ci.

18.11 Révoquer une paire de clés

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web `https://guide.boum.org/`.

Durée : 15 à 30 minutes.

Si notre clé privée était compromise, il faudrait faire parvenir le certificat de révocation créé précédemment à nos correspondants, pour que ceux-ci ne puissent plus l'utiliser et sachent qu'elle n'est plus de confiance.

page 134

Attention : les instructions qui suivent révoqueront de manière irréversible notre clé. À utiliser avec modération !

18.11.1 Faire savoir que notre paire de clés est compromise

Dans le cas ou notre propre paire de clés est compromise, par exemple si celle-ci a été obtenue après piratage de notre système, l'enjeu est d'arriver à le faire savoir à nos correspondants.

Que ce soit sous *Tails* ou avec une Debian chiffrée, il faudra tout d'abord importer ce certificat de révocation.

Importer le certificat de révocation

page 131 Contrairement à l'importation d'une clé il nous faudra le faire en utilisant un terminal.

tome 1 ch. 11 On va commencer la commande en tapant (**sans** faire *Entrée*) :

```
gpg --import
```

Puis on fera glisser le fichier du certificat de révocation dans le terminal pour obtenir quelque chose ressemblant à :

```
gpg --import '/home/amnesia/Certificat de révocation pour la clé 2A544427.rev
```

Taper ensuite sur entrée, le terminal doit renvoyer en retour un message ressemblant à :

```
gpg :clef 2A544427  :« Alice (exemple seulement) <alice@example.org> » certificat de
↪    révocation importé
gpg :      Quantité totale traitée  :1
gpg :nouvelles révocations de clef  :1
gpg :3 marginale(s) nécessaire(s), 1 complète(s) nécessaire(s),
     modèle de confiance PGP
gpg :profondeur  :0  valables  :  1  signées  :  0
     confiance  :0 i., 0 n.d., 0 j., 0 m., 0 t., 1 u.
```

Maintenant, il nous reste encore à faire savoir à nos correspondants que notre paire de clés a été compromise, car pour l'instant seul notre ordinateur est *au courant*. Pour y remédier nous pouvons publier notre clé désormais révoquée. Il faudra ensuite dire à nos correspondants, par exemple par email, qu'il leur faut se synchroniser avec le serveur de clés afin de révoquer notre clé publique en leur possession. Si en revanche nous n'avons pas publié notre clé publique sur un serveur de clés, il faudra inclure le certificat de révocation dans un email envoyé à nos correspondants.

Publier la paire de clés révoquée

Si notre clé publique a au préalable été publiée sur un serveur de clés, le mieux est de se synchroniser avec ce même serveur, pour que notre clé publique y soit désormais révoquée également, permettant ainsi à tous nos correspondants d'en être avertis en se synchronisant également. Attention toutefois, si aucune clé n'est sélectionnée, c'est l'intégralité du trousseau qui va être envoyé au serveur de clés. page 134

Pour cela, que ce soit sous *Tails* ou dans une Debian, lancer *Mots de passe et clés* à partir du menu *Applications* → *Outils système* → *Préférences* et suivre la partie *Publier sa clé publique sur les serveurs de clés* de l'outil créer et maintenir une paire de clés. page 134

Une fois cette synchronisation effectuée, reste à faire savoir cela à nos correspondants. Pas de recette toute faite pour ça, entre envoyer un email chiffré à ceux-ci, le leur faire savoir de vive-voix, *etc.*

18.11.2 Révoquer la paire de clés d'un correspondant

Si l'un de nos correspondant nous a fait savoir que sa paire de clés est compromise et qu'il l'a révoquée, il nous faut mettre cela à jour sur notre ordinateur, que ce soit *Tails* ou une Debian chiffrée.

Se synchroniser avec un serveur de clés

Dans le cas où notre correspondant a mis à jour sur un serveur de clés, sa clé publique désormais révoquée, il nous faudra simplement se synchroniser avec ce serveur de clés. Pour cela, lancer *Mots de passe et clés* à partir du menu *Applications* → *Outils système* → *Préférences*. Sélectionner la clé que l'on veut synchroniser, puis cliquer sur le menu *Distant* puis *Synchroniser et publier des clés*. Si aucun serveur de clés n'a été choisi, cliquer sur le bouton *Serveurs de clés*, et selectionner `hkp://pool.sks-keyservers.net` pour publier nos clés. Fermer la fenêtre et cliquer enfin sur *Synchroniser*.

Importer le certificat de révocation d'un correspondant

Si par contre la clé compromise de notre correspondant n'est pas disponible sur un serveur de clés, ou non synchronisée, et que celui-ci nous a fait parvenir le certificat de révocation, il nous faudra l'importer nous-mêmes. Pour cela, suivre les étapes du paragraphe *Importer le certificat de révocation* précédent.

Chapitre 19

Utiliser la messagerie instantanée avec OTR

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : une demi-heure à une heure.

L'objectif de cet outil est de dialoguer avec une personne en utilisant la messagerie instantanée avec chiffrement et authentification. On va pour cela utiliser le protocole OTR [1] qui permet d'ajouter chiffrement, authentification et confidentialité persistante [2] à nombre de protocoles de messagerie instantanée.

19.1 Installer le client de messagerie instantanée Pidgin

On va utiliser pour cela le client de messagerie *Pidgin*. En effet, il dispose d'une bonne prise en charge du chiffrement OTR. De plus, il permet d'utiliser différents protocoles de messagerie instantanée, comme *XMPP*, *IRC* ou encore *Yahoo ! Messenger*[3]. Ce logiciel est installé dans le système *live Tails*, mais seuls les protocoles XMPP et IRC y sont pris en charge, les autres étant difficiles à anonymiser. Sur une Debian chiffrée, il faudra commencer par installer les paquets `pidgin` ainsi que `pidgin-otr`.

tome 1 ch. 15
tome 1 § 16.3

19.2 Lancer Pidgin

Pour ouvrir le logiciel de messagerie instantanée, cliquer sur *Messagerie internet Pidgin* à partir du menu *Applications* → *Internet*.

19.3 Configurer un compte de messagerie

Lorsqu'on ouvre *Pidgin* et qu'aucun compte de messagerie n'est configuré, une fenêtre propose d'ajouter un nouveau compte.

Pour configurer un nouveau compte, cliquer sur le bouton *Ajouter…*.

1. Wikipédia, 2014, *Off-the-Record Messaging* [https://fr.wikipedia.org/wiki/Off-the-Record_Messaging]
2. La confidentialité persistante est une propriété en cryptographie qui garantit que la découverte par un adversaire de la clé privée d'un correspondant ne compromet pas la confidentialité d'une communication. (Wikipédia, 2014, *Confidentialité persistante* [https://fr.wikipedia.org/wiki/Confidentialité_Persistante]).
3. Pour une liste exhaustive des protocoles pris en charge par Pidgin, se référer à leur site web [http://www.pidgin.im/] (en anglais).

Une fenêtre *Ajouter un compte* s'ouvre. Si l'on dispose déjà d'un compte de messagerie instantanée, fournir les informations nécessaires concernant ce compte, en commençant par sélectionner le *Protocole* que l'on souhaite utiliser. Sinon en créer un au préalable.

19.4 Créer un compte de messagerie instantanée

tome 1 ch. 12

Si l'on ne dispose pas de compte de messagerie instantanée, c'est le moment d'en créer un. Tout comme pour un compte mail, un identifiant et une phrase de passe seront nécessaires, pour éviter d'utiliser tout le temps la même ou bien de risquer de l'oublier, il est possible d'utiliser un gestionnaire de mots de passe.

page 149

Certains fournisseurs d'adresses email, comme le collectif états-unien Riseup, proposent un compte de messagerie instantanée [4] à toute personne y disposant d'une adresse email tout comme un compte Facebook donne accès à la messagerie instantanée du site, Facebook Messenger.

On peut utiliser des serveurs communautaires où l'inscription est libre. Par exemple, une liste de serveurs XMPP [5] libres est disponible sur le site jabberfr.org [6]. Une fois un serveur choisi et les informations nécessaires [7] entrées dans la fenêtre de *Pidgin*, cocher la case *Créer ce nouveau compte sur le serveur.*

Il est aussi possible de se connecter à des serveurs du protocole IRC [8] sans disposer de compte [9].

19.5 Chiffrer la connexion au serveur XMPP

Par défaut, *Pidgin* configure le nouveau compte pour qu'il chiffre la communication avec le serveur XMPP. Si le certificat est bien signé par une Autorité de Certification, la connexion se déroulera sans problème, et *Pidgin* enregistrera le certificat du serveur XMPP dans sa configuration.

Si le certificat du serveur n'est pas signé, ou que pour une raison ou une autre *Pidgin* n'arrive pas à vérifier son authenticité, il est alors nécessaire de mettre en place les mêmes techniques que lors de l'installation d'un certificat dans son navigateur web pour le vérifier, sans quoi un aversaire pourrait usurper l'identité du serveur.

page 121

page 64

Dans ce cas, lors de votre première connexion, *Pidgin* affichera une fenêtre demandant si l'on veut *Accepter le certificat pour ?* Il expliquera également la raison pour laquelle il n'a pas voulu accepter le certificat (*Le certificat est auto-signé. Il ne peut être vérifié automatiquement.* si par exemple le certificat n'est pas signé par une Autorité de Certification). En cliquant sur *Voir le certificat…*, *Pidgin* affichera l'empreinte numérique de celui-ci, vous permettant de le vérifier.

tome 1 § 5.2

page 121

19.6 Activer le plugin *Off-the-Record*

Dans le menu *Outils*, cliquer sur *Plugins*. Trouver la ligne « Messagerie confidentielle 'Off-the-Record' » et cocher la case correspondante pour activer le plugin. Il est possible en cliquant sur *Configurer le plugin* de choisir certaines options telles que *Ne pas archiver les conversations d'OTR.*

4. riseup.net, 2013, *Riseup chat* [https://help.riseup.net/en/chat] (en anglais).
5. Wikipédia, 2014, *Extensible Messaging and Presence Protocol* [https://fr.wikipedia.org/wiki/XMPP]
6. Liste de serveurs XMPP communautaires [http://wiki.jabberfr.org/w/index.php?title=index.php&title=Serveurs]
7. Pour plus de détails sur les informations à renseigner pour créer un compte XMPP, voir le site de Linuxpedia [http://www.linuxpedia.fr/doku.php/internet/pidgin-jabber]
8. Wikipédia, 2014, *Internet Relay Chat* [https://fr.wikipedia.org/wiki/Internet_Relay_Chat]
9. irchelp.org, 2012, *IRC Networks and Server Lists* [http://www.irchelp.org/irchelp/networks/] (en anglais).

19.7 Mettre en place une conversation privée

19.7.1 Commencer une conversation privée

Pour commencer une conversation privée, double-cliquer sur un nom se trouvant dans la colonne de droite de la fenêtre d'un salon de discussion où l'on se trouve. Une fenêtre de conversation s'ouvre. Cliquer alors sur le menu *OTR → Commencer une conversation privée*.

Si c'est la première fois qu'on utilise OTR avec ce compte, *Pidgin* va alors générer une clé privée et afficher une fenêtre *Génération de la clé privée*. Cette clé est unique pour un compte donné. Si l'on possède plusieurs comptes de messagerie instantanée, on aura donc plusieurs clés. Lorsqu'elle affiche que la génération de cette clé est *effectuée*, on peut fermer cette fenêtre en cliquant sur *Valider*.

Pidgin affiche alors *Alice n'a pas encore été authentifiée. Vous devriez authentifier ce contact*. Cela signifie que notre conversation est chiffrée, mais qu'un adversaire pourrait se faire passer pour Alice. Pour être sûr de parler avec Alice, il faut l'authentifier. page 64

19.7.2 Authentifier un correspondant

Pour authentifier un correspondant, il faut soit s'être mis d'accord au préalable sur un secret, soit disposer d'un moyen de communication autre que la messagerie instantanée, que l'on considère comme sûr. Ce moyen peut être une conversation de vive-voix, un email chiffré, *etc.*

OTR propose trois façons d'authentifier un contact :

- par question-réponse : on définit une question et sa réponse. La question étant ensuite posée à notre correspondant ;
- avec un secret partagé : un secret connu uniquement des deux interlocuteurs est demandé afin de vérifier qu'on dialogue bien avec la personne escomptée ;
- grâce à la vérification manuelle de l'empreinte : on vérifie que l'empreinte de la clé de la personne avec qui l'on s'apprête à avoir une conversation chiffrée est la même que celle qui nous a été fournie par un moyen *authentifié*.

Une fois les secrets, les questions-réponses ou les empreintes échangés, cliquer sur le menu *OTR → Authentifier le contact*. Choisir la méthode d'authentification en-dessous de *Comment désirez-vous authentifier votre contact ?*, puis répondre aux questions. Enfin, cliquer sur *Authentifier*.

Si l'authentification est réussie, le statut de la conversation devient *Privé*, ce qui signifie qu'elle est non seulement chiffrée, mais aussi authentifiée.

Si l'on utilise un système non-live ou que l'on a activé la persistance de *Pidgin* dans *Tails*, cette étape d'authentification n'est à effectuer qu'une fois pour toutes pour un contact donné. tome 1 § 14.5

19.7.3 Terminer une conversation

Une fois notre dialogue terminé, cliquer sur le menu *OTR → Terminer la conversation privée*. Cela efface la clé de chiffrement temporaire générée pour cette conversation de la mémoire vive de l'ordinateur. Même si un adversaire obtenait nos clés privées, il lui serait alors impossible de déchiffrer la conversation *a posteriori*.

Chapitre 20

Gérer des mots de passe

Les logiciels évoluent, c'est pourquoi il est vivement conseillé d'utiliser la version la plus à jour de cet outil, qui est disponible sur le site web https://guide.boum.org/.

Durée : 15 à 30 minutes.

Lorsqu'on crée une adresse email, un compte sur un site web, *etc.* ce compte est en général protégé par un mot de passe.

Il est important de ne pas utiliser le même mot de passe pour des accès à des services ayant des niveaux de sécurité différents, par exemple une boîte mail et un compte sur un site de jeu d'échecs en ligne.

Il est important aussi de ne pas utiliser le même mot de passe pour des identités contextuelles différentes, afin que la compromission de l'une d'entre elles n'entraîne pas la compromission des autres.

page 53

Il existe deux bonnes écoles pour choisir de bons mots de passe :

- choisir et retenir une bonne phrase de passe ;
- utiliser de bons mots de passe générés aléatoirement et les enregistrer dans un gestionnaire de mots de passe qui, lui, est protégé par une bonne phrase de passe que l'on retiendra.

20.1 Choisir une bonne phrase de passe

La première école a l'avantage de ne nécessiter aucun support de stockage : on a toujours ses phrases de passe avec soi. Pour l'appliquer, consulter choisir une bonne phrase de passe.

tome 1 ch. 12

Toutefois, lorsqu'on multiplie les comptes ainsi que les identités contextuelles, cela peut faire beaucoup de phrases de passe à retenir.

20.2 Utiliser un gestionnaire de mots de passe

La seconde méthode peut alors nous être utile. Dans la pratique, on aura une phrase de passe à retenir par identité, notre gestionnaire de mots de passe se chargeant ensuite de conserver les différents mots de passe liés à cette identité. Cela peut se faire sur un système Debian chiffré comme sur un système *live* amnesique en utilisant la persistance.

tome 1 ch. 15
tome 1 ch. 14
tome 1 § 14.5

20.2.1 Installer le gestionnaire de mots de passe

tome 1 § 16.3

On va utiliser le gestionnaire de mots de passe *KeePassX*. S'il n'est pas installé sur notre système, installer le paquet `keepassx`. Le logiciel *KeePassX* est installé par défaut dans *Tails*.

20.2.2 Lancer KeePassX

Pour lancer *KeePassX*, choisir *Applications* → *Accessoires* → *KeePassX*.

20.2.3 Créer et enregistrer une base de données de mots de passe

Une base de données de mots de passe est un ensemble de mots de passe qui seront stockés dans une même base de données *KeePassX* et chiffrés par la phrase de passe associée.

tome 1 § 14.5

Si on choisit d'utiliser *KeePassX* dans *Tails*, il faudra au préalable activer la persistance et activer l'option *Données personnelles*.

Il faut tout d'abord créer une nouvelle base de données de mots de passe et l'enregistrer pour l'utiliser lors de futures sessions de travail. Pour créer une nouvelle base de données de mots de passe, choisir *Fichier* puis *Nouvelle base de données…*

Une fenêtre apparaît et demande de définir la clé maître. Il s'agit de la phrase de passe servant à déchiffrer la base de donnée de mots de passe. Spécifier une phrase de passe dans la boîte de texte *Mot de passe*, puis cliquer sur *OK*. Taper cette phrase de passe de nouveau dans la fenêtre suivante, puis cliquer sur *OK*.

Pour stocker la base de données de mots de passe nouvellement créée afin de l'utiliser lors de prochaines sessions de travail, choisir *Fichier* puis *Enregistrer la base de données*. Taper `keepassx` dans le champ *Nom*. Si l'on utilise *Tails*, sélectionner *Persistent* dans la liste des dossiers du menu déroulant de gauche. Sinon, garder le choix par défaut. Cliquer sur *Enregistrer*.

20.2.4 Générer et enregistrer un mot de passe aléatoire

KeePassX permet également de générer des mots de passe aléatoires plus robustes que des mots de passe dont on pourrait se souvenir.

Dans *KeePassX*, cliquer sur *Entrées* puis *Ajouter une nouvelle entrée…*. Remplir les champs utiles. Arrivé au champ *Mot de passe*, cliquer sur le bouton *Gen.*.

Une boîte de dialogue *Générateur de mots de passe* s'ouvre. Choisir parmi les options disponibles et cliquer sur *Générer*, puis sur *OK*.

Il est préférable d'utiliser des lettres minuscules, majuscules et des chiffres, puis d'augmenter le nombre de caractères du mot de passe (au minimum 32), puisqu'on aura pas à retenir ce dernier.

Une fois revenu à la boîte de dialogue *Nouvelle Entrée*, cliquer sur *OK*.

Cliquer alors sur *Fichier* puis *Enregistrer la base de données*.

20.2.5 Restaurer et déverrouiller la base de données de mots de passe

Lorsqu'on veut utiliser une base de données de mots de passe préalablement enregistrée, il nous faut la déverrouiller. Pour cela, lancer *KeePassX*. Si une base de données de mots de passe est trouvée automatiquement, une fenêtre s'ouvre demandant d'*Entrer la clé maître* afin de déverrouiller la base de données. Taper la phrase de passe associée à la base de données que l'on souhaite déverrouiller et cliquer sur

OK. Sinon, localiser la base de données à partir de *Fichier* puis *Ouvrir une base de données…*.

Si vous entrez une mauvaise phrase de passe, le message d'erreur suivant apparaît :

L'erreur suivante est survenue lors de l'ouverture de la base de données :

Le test de hachage a échoué.

La clé est mauvaise ou le fichier est endommagé.

Cliquer sur *OK* et essayer de nouveau.

20.2.6 Utiliser un mot de passe enregistré

Après avoir restauré et déverrouillé la base de données de mots de passe, on peut utiliser les mots de passe qui y sont enregistrés.

Pour utiliser un identifiant enregistré, le sélectionner dans la liste. Aller dans la fenêtre où l'on souhaite l'utiliser et placer le curseur dans le champ d'entrée. Retourner alors à la fenêtre de *KeePassX* puis cliquer sur *Exécuter la Saisie Automatique* à partir du menu *Entrées*. *KeePassX* se réduit alors automatiquement dans le tableau de bord.

Attention : La saisie automatique permet aussi de faire de belles boulettes, comme coller son mot de passe dans une fenêtre de messagerie instantanée... et envoyer le message automatiquement. Il faut donc faire très attention à l'endroit où on place le curseur avant d'exécuter la saisie automatique.

Il est possible que cette méthode de saisie automatique ne fonctionne pas pour tous les types d'interfaces. Dans ce cas, faire clic-droit sur l'identifiant sélectionné précédemment, et choisir *Copier l'utilisateur dans le presse-papier*, faire ensuite clic-droit et *Coller* à l'endroit où saisir le nom d'utilisateur. Pour copier le mot de passe, refaites un clic-droit sur l'identifiant dans la fenêtre de *KeePassX*, et cliquer sur *Copier le mot de passe vers le presse-papier*, puis le coller dans le champ d'entrée de mot de passe.

Index

A
administrateurs, *voir* admins
admins, 16
adresse IP, 13, 15
adresse MAC, *voir* adresse matérielle
adresse matérielle, 10
adresse privée, 15
adresse publique, 15
ADSL, 10
anonymity set, 73
ARPANET, 9
AS, *voir* système autonome

B
backbone, *voir* épine dorsale
box, 15
bridge, *voir* switch

C
carte réseau, 10
Claws Mail, *voir* client mail
client de messagerie, *voir* client mail
client mail, 94
commutateur, *voir* switch
Cookie, 26
Cookie Flash, *voir* Local Shared Object

D
deep packet inspection, *voir* examen approfondi des paquets
DHCP, 15
DNS, *voir* nom de domaine
domaine de premier niveau, 44
DPI, *voir* examen approfondi des paquets, 41

E
en-tête, 28
encapsulation, 12
épine dorsale, 18
Ethernet, 10
examen approfondi des paquets, 29, 45

F
Facebook, 35
FAI, *voir* fournisseur d'accès à Internet
fibre optique, 10
filoutage, *voir* hameçonnage
filtrage, 45
firewall, *voir* pare-feu
Flash, 26
fournisseur d'accès à Internet, 15

G
Google, 35

H
hameçonnage, 43
HTTP, 93
HTTPS, 93

I
IMAP, 20, 93
IMAPS, 20, 93
Internet Protocol, 12
interopérabilité, 11
IP, *voir* Internet Protocol
IPv4, *voir* Internet Protocol
IPv6, *voir* Internet Protocol
IRC, 20

L
LAN, *voir* réseau local
Local Shared Object, 26
LSO, *voir* Local Shared Object

M
MAC, *voir* adresse matérielle
modem, 10, 15

N
NAT, 15
nom de domaine, 22
nom de domaine de premier niveau, *voir* domaine de premier niveau

O
Outlook, *voir* client mail

P
pare-feu, 21
passerelle, 15
phishing, *voir* hameçonnage
point d'accès, 14
pont, *voir* switch
POP, 20, 93
POPS, 20, 93
port, 20
portail captif, 28
protocole applicatif, 20
protocole IP, *voir* Internet Protocol
protocole réseau, 12

R
radio, *voir* Wi-Fi
réseau local, 14
RJ-45, *voir* Ethernet
robot, 98
routeur, 15, 17, 18, 28

S
site mirroir, 42
Skype, 20
SMTP, 20, 93
SMTPS, 20, 93
spam, 98
switch, 14
système autonome, 16

T
TCP, 13
Thunderbird, *voir* client mail
TLD, *voir* domaine de premier niveau
top level domain, *voir* domaine de premier niveau

U
UDP, 13

W
webmail, 93
Wi-Fi, 10, 14

X
XMPP, 20

Photo page 11 de David Monniaux, licence CC BY-SA 3.0, trouvée sur :
https://commons.wikimedia.org/wiki/File:Ethernet_RJ45_connector_p1160054.jpg.

Photo page 17 de Geek2003, licence CC BY-SA 3.0, trouvée sur :
https://commons.wikimedia.org/wiki/File:Avaya_Secure_Router_2330.jpg.

Photo page 23 de Victor Grigas, licence CC BY-SA 3.0, trouvée sur :
https://commons.wikimedia.org/wiki/File:
Wikimedia_Foundation_Servers-8055_08.jpg.

Les autres schémas sont faits par les auteurs du guide et utilisent des icônes : de GNOME Project, licence CC BY-SA 3.0 ; de Silvestre Herrera, licence GPLv2 trouvées sur http://www.silvestre.com.ar/ ; du domaine public trouvées sur http://openclipart.org.

guide d'autodéfense numérique

tome 2

en ligne

[…] *nous n'avons pas envie d'être contrôlables par quelque « Big Brother » que ce soit. Qu'il existe déjà ou que l'on anticipe son émergence, le mieux est sans doute de faire en sorte qu'il ne puisse pas utiliser, contre nous, tous ces merveilleux outils que nous offrent — ou que lui offrent — les technologies numériques.* […]

Même si l'on choisit de ne pas les utiliser directement, d'autres le font pour nous. Alors, autant essayer de comprendre ce que ça implique.

Face à ces constats, la seule voie praticable semble être de devenir capables d'imaginer et de mettre en place des politiques de sécurité adéquates.
Tout l'enjeu de ce guide est de fournir cartes, sextant et boussole à quiconque veut cheminer sur cette route.

C'est l'objet principal de ce second tome que de permettre à tout un chacun de comprendre quels sont les risques et les limites associés à l'utilisation d'Internet [et] *de se donner les moyens de faire des choix éclairés quant à nos usages de l'Internet.*

Un livre à lire, relire, pratiquer, en solitaire ou à plusieurs, à faire découvrir et à partager… ou comment affiner l'art de la navigation dans les eaux troubles du monde numérique.